

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant(s): MOGI, et al.  
Serial No.: Not yet assigned  
Filed: August 5, 2003  
Title: COMPUTER SYSTEM FOR MANAGING PERFORMANCES OF  
STORAGE APPARATUS AND PERFORMANCE  
MANAGEMENT METHOD OF THE COMPUTER SYSTEM  
Group: Not yet assigned

LETTER CLAIMING RIGHT OF PRIORITY

Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

August 5, 2003

Sir:

Under the provisions of 35 USC 119 and 37 CFR 1.55, the applicant(s) hereby claim(s) the right of priority based on Japanese Patent Application No.(s) 2003-093544, filed March 31, 2003.

A certified copy of said Japanese Application is attached.

Respectfully submitted,

ANTONELLI, TERRY, STOUT & KRAUS, LLP

  
\_\_\_\_\_  
Carl I. Brundidge  
Registration No. 29,621

CIB/alb  
Attachment  
(703) 312-6600

日 本 国 特 許 庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出 願 年 月 日

Date of Application:

2003年 3月31日

出 願 番 号

Application Number:

特願2003-093544

[ ST.10/C ]:

[ JP 2003-093544 ]

出 願 人

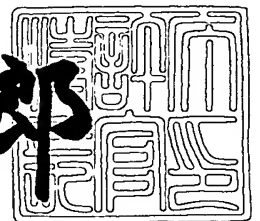
Applicant(s):

株式会社日立製作所

2003年 6月25日

特 許 庁 長 官  
Commissioner,  
Japan Patent Office

太田信一郎



出証番号 出証特2003-3050083

【書類名】 特許願  
【整理番号】 NT03P0186  
【提出日】 平成15年 3月31日  
【あて先】 特許庁長官 殿  
【国際特許分類】 G06F 3/06

G06F 12/00571

G06F 13/12

G06F 17/30

【発明者】

【住所又は居所】 神奈川県川崎市麻生区王禅寺1 0 9 9 番地 株式会社日立製作所 システム開発研究所内

【氏名】 茂木 和彦

【発明者】

【住所又は居所】 神奈川県川崎市麻生区王禅寺1 0 9 9 番地 株式会社日立製作所 システム開発研究所内

【氏名】 西川 記史

【発明者】

【住所又は居所】 神奈川県川崎市麻生区王禅寺1 0 9 9 番地 株式会社日立製作所 システム開発研究所内

【氏名】 出射 英臣

【特許出願人】

【識別番号】 000005108

【氏名又は名称】 株式会社日立製作所

【代理人】

【識別番号】 100068504

【弁理士】

【氏名又は名称】 小川 勝男

【電話番号】 03-3661-0071

【選任した代理人】

【識別番号】 100086656

【弁理士】

【氏名又は名称】 田中 恭助

【電話番号】 03-3661-0071

【選任した代理人】

【識別番号】 100094352

【弁理士】

【氏名又は名称】 佐々木 孝

【電話番号】 03-3661-0071

【手数料の表示】

【予納台帳番号】 081423

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【プルーフの要否】 要



【書類名】 明細書

【発明の名称】 計算機システムの性能管理方法、および、記憶装置の性能を管理する計算機システム

【特許請求の範囲】

【請求項 1】 プログラムを実行する計算機と、そのプログラムの実行のためのデータを記憶する記憶装置とを有する計算機システムの性能管理方法において、

(1) 前記プログラムに要求される性能の情報を取得する手順、

(2) 前記プログラムを実行するためのデータのマッピング情報を、前記計算機、前記記憶装置から取得する手順、

(3) 前記記憶装置から稼動統計情報を取得する手順、

(10) 前記(1)ないし(3)の手順で取得された情報を用いて前記記憶装置の設定変更案を作成する手順、

(11) 前記記憶装置に対して前記設定変更案に従った設定変更を指示する手順を有することを特徴とする性能管理方法。

【請求項 2】 さらに、

(4) 前記プログラムの優先度に関する情報を取得する手順を有し、

前記(10)の手順で、前記(4)の手順で取得した情報をも利用して前記記憶装置の設定変更案を作成することを特徴とする請求項 1 記載の計算機システムの性能管理方法。

【請求項 3】 前記(1)の手順が、前記プログラムに要求される処理実行時間と、処理実行時間見積り方法と、前記プログラムに入力として与えられるデータのデータ量とを取得する処理を含み、

前記(10)の手順で、前記記憶装置の設定変更案を作成する際して、前記プログラムの処理実行時間と、前記処理実行時間見積り方法と、前記プログラムへの入力データ量とを利用することを特徴とする請求項 1 記載の計算機システムの性能管理方法。

【請求項 4】 前記処理実行時間見積り方法を取得する際に、前記(3)の手順で取得した前記稼動統計情報を利用することを特徴とする請求項 3 記載の計

算機システムの性能管理方法。

【請求項 5】 前記（１）の手順が、前記プログラムに要求される処理の応答時間に関する情報と、前記プログラムの処理の応答時間の実測値に関する情報とを取得する処理を含み、

前記（１０）の手順で、前記記憶装置の設定変更案を作成する際して、前記プログラムに要求される処理の応答時間に関する情報と、前記プログラムの処理の応答時間の実測値に関する情報とを利用することを特徴とする請求項 1 記載の計算機システムの性能管理方法。

【請求項 6】 前記記憶装置が、外部装置と接続するためのポートを備え、そのポートからのアクセスの処理量を制限する優先アクセス制御機能を有する記憶装置である場合に、

前記（１０）の手順で作成する設定変更案が、前記優先アクセス制御機能の設定値の変更案であることを特徴とする請求項 1 記載の計算機システムの性能管理方法。

【請求項 7】 前記記憶装置が、データキャッシュを備え、そのデータキャッシュを幾つかの領域に区分された領域により管理して、その区分されたデータキャッシュの領域を動的に追加・削除・領域サイズ変更をする機能を有する記憶装置である場合に、

前記（１０）の手順で作成する設定変更案が、前記区分されたデータキャッシュの領域の追加・削除・領域サイズ変更案であることを特徴とする請求項 1 記載の計算機システムの性能管理方法。

【請求項 8】 前記記憶装置が、少なくとも二つ以上の物理記憶手段を備え

前記物理記憶手段の記憶領域を利用した論理記憶領域を提供する機能と、前記論理記憶領域の前記物理記憶手段の記憶領域への対応関係を動的に変更する物理記憶位置変更機能とを有し、

前記（１０）の手順で作成する設定変更案が、前記物理記憶位置変更機能による前記物理記憶手段と前記論理記憶領域の対応関係の変更指示案であることを作成することを特徴とする請求項 1 記載の計算機システムの性能管理方法。

【請求項 9】 データベース管理システムを動作させる第一の計算機と、前記データベース管理システムに対して処理要求を出すプログラムを実行する第二の計算機と、データベース管理システムで取扱うデータを記憶する記憶装置とを有する計算機システムの性能管理方法において、

(21) 前記プログラムに要求される性能の情報を取得する手順、

(22) 前記プログラムが前記データベース管理システムに対して要求する処理において利用するデータベースのデータの情報を取得する手順、

(23) 前記プログラムの実行するためのデータのマッピング情報を、前記第一の計算機、前記記憶装置から取得する手順、

(24) 前記記憶装置から稼動統計情報を取得する手順、

(30) 前記(21)ないし(24)の手順で取得された情報を用いて前記記憶装置の設定変更案を作成する手順、

(31) 前記記憶装置に対して前記設定変更案に従った設定変更を指示する手順を有することを特徴とする計算機システムの性能管理方法。

【請求項 10】 前記第一の計算機と前記第二の計算機が同一の計算機であることを特徴とする請求項 9 記載の計算機システムの性能管理方法。

【請求項 11】 前記記憶装置が、外部装置と接続するためのポートを備え、そのポートからのアクセスの処理量を制限する優先アクセス制御機能を有する記憶装置である場合に、

前記(30)の手順で作成する設定変更案が、前記優先アクセス制御機能の設定値の変更案であることを特徴とする請求項 9 記載の計算機システムの性能管理方法。

【請求項 12】 前記記憶装置が、データキャッシュを備え、そのデータキャッシュを幾つかの領域に区分された領域により管理して、その区分されたデータキャッシュの領域を動的に追加・削除・領域サイズ変更をする機能を有する記憶装置である場合に、

前記(30)の手順で作成する設定変更案が、前記区分されたデータキャッシュの領域の追加・削除・領域サイズ変更案であることを特徴とする請求項 9 記載の計算機システムの性能管理方法。

【請求項 1 3】 前記記憶装置が、少なくとも二つ以上の物理記憶手段を備え、

前記物理記憶手段の記憶領域を利用した論理記憶領域を提供する機能と、前記論理記憶領域の前記物理記憶手段の記憶領域への対応関係を動的に変更する物理記憶位置変更機能とを有する記憶装置である場合に、

前記（30）の手順で作成する設定変更案が、前記物理記憶位置変更機能による前記物理記憶手段と前記論理記憶領域の対応関係の変更指示案であることを作成することを特徴とする請求項 9 記載の計算機システムの性能管理方法。

【請求項 1 4】 データベース管理システムを動作させる第一の計算機と、前記データベース管理システムに対して処理要求を出すプログラムを実行する第二の計算機と、データベース管理システムで取扱うデータを記憶する記憶装置とを有する記憶装置の性能を管理する計算機システムにおいて、

管理装置を有し、

この管理装置が、前記プログラムに要求される性能の情報と、

前記プログラムが前記データベース管理システムに対して要求する処理において利用するデータベースのデータの情報と、

前記第一の計算機、前記記憶装置から得られる前記プログラムの実行するためのデータのマッピング情報と、

前記記憶装置から得られる稼動統計情報とを取得して、

前記管理装置が、取得した情報を用いて前記記憶装置の設定変更案を作成して

前記記憶装置に対して前記設定変更案に従った設定変更を指示することを特徴とする記憶装置の性能を管理する計算機システム。

【請求項 1 5】 前記第一の計算機と前記第二の計算機が同一の計算機であることを特徴とする請求項 1 4 記載の記憶装置の性能を管理する計算機システム。

【請求項 1 6】 さらに、

前記第一の計算機と、前記記憶装置との間に接続され、両者の間のデータ転送を制御する記憶制御装置を有することを特徴とする請求項 1 4 記載の記憶装置の

性能を管理する計算機システム。

【請求項 1 7】 前記管理装置の前記情報の収集と、前記設定変更案の作成と、前記設定変更案に従った設定変更の指示とをおこなう機能を、前記第一の計算機または前記第二の計算機で実現することを特徴とする請求項 1 4 記載の記憶装置の性能を管理する計算機システム。

【請求項 1 8】 前記管理装置の情報の収集と、前記設定変更案の作成と、前記設定変更案に従った設定変更を指示とをおこなう機能を、前記記憶装置で実現することを特徴とする請求項 1 4 記載の記憶装置の性能を管理する計算機システム。

【請求項 1 9】 前記管理装置の情報の収集と、前記設定変更案の作成と、前記設定変更案に従った設定変更を指示とをおこなう機能を、前記記憶制御装置で実現することを特徴とする請求項 1 6 記載の記憶装置の性能を管理する計算機システム。

【請求項 2 0】 データベース管理システムを動作させる第一の計算機と、前記データベース管理システムに対して処理要求を出すプログラムを実行する第二の計算機と、データベース管理システムで取扱うデータを記憶する記憶装置とを有する計算機システムの性能を管理する管理装置において、

この管理装置が、前記プログラムに要求される性能の情報と、

前記プログラムが前記データベース管理システムに対して要求する処理において利用するデータベースのデータの情報と、

前記第一の計算機、前記記憶装置から得られる前記プログラムの実行するためのデータのマッピング情報と、

前記記憶装置から得られる稼動統計情報とを取得して、

取得した情報を用いて前記記憶装置の設定変更案を作成して、

前記記憶装置に対して前記設定変更案に従った設定変更を指示することを特徴とする計算機システムの性能を管理する管理装置。

【請求項 2 1】 前記第一の計算機と前記第二の計算機が同一の計算機であることを特徴とする請求項 2 0 記載の計算機システムの性能を管理する管理装置。

【請求項 2 2】 前記記憶装置が、外部装置と接続するためのポートを備え、そのポートからのアクセスの処理量を制限する優先アクセス制御機能を有する記憶装置であって、

前記管理装置が作成する設定変更案が、前記優先アクセス制御機能の設定値の変更案であることを特徴とする請求項 2 0 記載の計算機システムの性能を管理する管理装置。

【請求項 2 3】 前記記憶装置が、データキャッシュを備え、そのデータキャッシュを幾つかの領域に区分された領域により管理して、その区分されたデータキャッシュの領域を動的に追加・削除・領域サイズ変更をする機能を有する記憶装置であって、

前記管理装置が作成する設定変更案が、前記区分されたデータキャッシュの領域の追加・削除・領域サイズ変更案であることを特徴とする請求項 2 0 記載の計算機システムの性能を管理する管理装置。

【請求項 2 4】 前記記憶装置が、少なくとも二つ以上の物理記憶手段を備え、

前記物理記憶手段の記憶領域を利用した論理記憶領域を提供する機能と、前記論理記憶領域の前記物理記憶手段の記憶領域への対応関係を動的に変更する物理記憶位置変更機能とを有する記憶装置であって、

前記管理装置で作成する設定変更案が、前記物理記憶位置変更機能による前記物理記憶手段と前記論理記憶領域の対応関係の変更指示案であることを特徴とする請求項 2 0 記載の計算機システムの性能を管理する管理装置。

【請求項 2 5】 プログラムを実行する計算機と、そのプログラムの実行のためのデータを記憶する記憶装置とを有する計算機システムの性能管理プログラムにおいて、

前記プログラムに要求される性能の情報を取得する機能と、

前記プログラムを実行するためのデータのマッピング情報を、前記計算機、前記記憶装置から取得する機能と、

前記記憶装置から稼動統計情報を取得する機能とにより、

前記計算機システムより情報を収集し、前記収集された情報を用いて前記記憶

装置の設定変更案を作成し、前記記憶装置に対して前記設定変更案に従った設定変更を指示する機能を実現することを特徴とする計算機システムの性能管理プログラム。

【請求項 2 6】 さらに、

前記プログラムの優先度に関する情報を取得する機能を有し、

前記計算機システムにより、プログラムの優先度に関する情報をも収集し、その情報をも利用して前記記憶装置の設定変更案を作成することを特徴とする請求項 2 5 記載の計算機システムの性能管理プログラム。

【請求項 2 7】 前記プログラムに要求される性能の情報を取得する機能では、前記プログラムに要求される処理実行時間と、処理実行時間見積り方法と、前記プログラムに入力として与えられるデータのデータ量とを取得して、

前記記憶装置の設定変更案を作成する際して、前記プログラムの処理実行時間と、前記処理実行時間見積り方法と、前記プログラムへの入力データ量とを利用することを特徴とする請求項 2 5 記載の計算機システムの性能管理プログラム。

【請求項 2 8】 前記処理実行時間見積り方法を取得する際に、前記機能で取得した前記稼動統計情報を利用することを特徴とする請求項 2 7 記載の計算機システムの性能管理プログラム。

【請求項 2 9】 前記プログラムに要求される性能の情報を取得する機能では、前記プログラムに要求される処理の応答時間に関する情報と、前記プログラムの処理の応答時間の実測値に関する情報とを取得して

前記記憶装置の設定変更案を作成する際して、前記プログラムに要求される処理の応答時間に関する情報と、前記プログラムの処理の応答時間の実測値に関する情報とを利用することを特徴とする請求項 2 5 記載の計算機システムの性能管理プログラム。

【請求項 3 0】 前記請求項 2 5 記載の計算機システムの性能管理プログラムを記録する計算機読み取り可能な記憶媒体。

【請求項 3 1】 データベース管理システムを動作させる第一の計算機と、前記データベース管理システムに対して処理要求を出すプログラムを実行する第二の計算機と、データベース管理システムで取扱うデータを記憶する記憶装置とを

有し、さらに、前記第一の計算機と、前記記憶装置との間に接続され、両者の間のデータ転送を制御する記憶制御装置とを有する計算機システムの性能管理プログラムにおいて、

前記プログラムに要求される性能の情報を取得する機能と、

前記プログラムが前記データベース管理システムに対して要求する処理において利用するデータベースのデータの情報を取得する機能と、

前記プログラムの実行するためのデータのマッピング情報を、前記第一の計算機、前記記憶装置から取得する機能と、

前記記憶装置から稼動統計情報を取得する機能とにより、

前記計算機システムより情報を収集し、前記収集された情報を用いて前記記憶装置の設定変更案を作成し、前記記憶装置に対して前記設定変更案に従った設定変更を指示する機能を実現することを特徴とする計算機システムの性能管理プログラム。

【請求項 3 2】 前記第一の計算機と前記第二の計算機が同一の計算機であることを特徴とする請求項 3 1 記載の計算機システムの性能管理プログラム。

【請求項 3 3】 前記管理装置の前記情報の収集と、前記設定変更案の作成と、前記設定変更案に従った設定変更の指示とをおこなう機能を、前記第一の計算機または前記第二の計算機で実現することを特徴とする請求項 3 1 記載の計算機システムの性能管理プログラム。

【請求項 3 4】 前記管理装置の情報の収集と、前記設定変更案の作成と、前記設定変更案に従った設定変更を指示とをおこなう機能を、前記記憶装置で実現することを特徴とする請求項 3 1 記載の計算機システムの性能管理プログラム。

【請求項 3 5】 前記管理装置の情報の収集と、前記設定変更案の作成と、前記設定変更案に従った設定変更を指示とをおこなう機能を、前記記憶制御装置で実現することを特徴とする請求項 3 1 記載の計算機システムの性能管理プログラム。

【請求項 3 6】 前記請求項 3 1 記載の計算機システムの性能管理プログラムを記録する計算機読み取り可能な記憶媒体。



## 【発明の詳細な説明】

## 【0001】

## 【発明の属する技術分野】

本発明は、計算機システムの性能管理方法、および、記憶装置の性能を管理する計算機システムに係り、特に、業務プログラム処理の要求性能に応じて、大規模データベース管理システム（DBMS, Database Management System）が稼動する計算機システム内の記憶装置の性能を調整し、要求されたシステムの性能を発揮するために用いて好適な計算機システムの性能管理方法、および、記憶装置の性能を管理する計算機システムに関する。

## 【0002】

## 【従来技術】

近年、データベース（DB, Database）に関する一連の処理・管理をおこなうソフトウェアであるデータベース管理システム（DBMS）は極めて重要なものとなってきており、サーバ計算機上で実行されるアプリケーションプログラム（APプログラム）のソフトウェア構成を調べても、DBを基盤として利用するものが多数を占めるようになってきている。

## 【0003】

APプログラムの利用に際しては、業務毎に定まる要求性能が存在し、それを守ることが強く求められており、その管理は計算機システムの管理業務の中で極めて重要である。一般に、DBMSの性能は、データへのアクセス性能から多大な影響を受ける。データアクセス性能は、データアクセスパスに含まれるハードウェアリソースの稼動状況に強い影響を受けるため、記憶装置の稼動状況を考慮した性能管理が重要である。

## 【0004】

特許文献1では、データアクセスパスに含まれるハードウェアリソースの性能解析を容易化するため、DBMSから記憶装置間のデータのマッピングを取得し、各レイヤにおける稼動状況（稼働率）や、ある表やファイル等、データの管理構造に対応する他のレイヤの管理構造の稼動状況を提示する技術を開示している。特定の部位が過負荷状態の場合には、そこが性能低下の原因となっている可能

性が極めて高い。そこで、特許文献1では、ある管理構造に属するデータの記憶位置を変更することによりそれを解消する最適化案の作成することになっている。

## 【0005】

一方、ハードウェアリソースの過負荷状態を解消することにより性能を最適化する機能を有する記憶装置が存在する。特許文献2は、計算機に論理記憶装置を認識させ、そのデータを記憶装置内部に複数存在する物理ディスク装置にマッピングして記憶する記憶装置において、論理記憶装置の物理ディスク装置へのマッピングを動的に（以下「他の処理を停止させることなく実施される」の意味で利用）変更することにより記憶装置のアクセス性能を向上する技術を開示している。この特許文献2では、稼働率が高い（過負荷状態の）物理ディスク装置に記憶されているデータの一部を他の物理ディスク装置に移動することにより、特定の物理ディスク装置が過負荷状態にならないようにし、性能を最適化する。

## 【0006】

また、Fibre Channel等を用いて、同一のポートを経由して複数の計算機と接続する場合に、特定の計算機からのアクセスを優先的に実行する機能を有する記憶装置が存在する（例えば、非特許文献1記載のPriority Access機能）。

## 【0007】

さらに、記憶装置の多くはキャッシュメモリを有している。したがって、データリード時にデータがキャッシュに存在している状態（ヒット）である確率を向上させることによりアクセス性能を向上させることができる。特許文献3では、データ種類毎に区分されたキャッシュ領域を持つ場合に、その領域割り当てを最適化してキャッシュヒット率を向上させる技術について開示している。この特許文献3では、LRU (Least Recently Used) 置換アルゴリズムによるキャッシュデータ置換制御をおこなうシステムで、キャッシュヒット時にLRU管理リスト中のどの位置に存在するかの情報を取得し、その情報を用いて領域割り当て変更時のキャッシュヒット率の見積りをおこない、領域分割の最適化をおこなっている。

## 【0008】

【特許文献1】

米国特許第6,035,306号明細書

【特許文献2】

特開平9-274544号公報

【特許文献3】

米国特許第5,434,992号明細書

【非特許文献1】

Hitachi Data Systems社,「Hitachi Freedom Storage(TM) Lightning 9900(TM) V Series」,pp.8-9,DISK-432-00b,2002年10月

【0009】

【発明が解決しようとする課題】

ユーザの業務における性能指標は、一括した大量データ処理をおこなうバッチ業務の場合には処理時間、オンライン業務の場合には応答時間や処理のスループットである場合が多い。しかしながら、従来技術が利用・提供する性能指標は、計算機システム内リソースの稼働率やキャッシュヒット率等の物理動作に直結したものであり、ユーザ視点での性能指標に直接結びつくものではない。また、多くの場合、従来技術が提供する複数の性能指標を調べ、その中から性能ボトルネックを発見・解消しないとユーザが求める性能を達成することはできない。現状では、それらの埋め合わせは管理者がおこなっており、しかも、これは簡単なものではなく、特別な技術を要する業務となっている。

【0010】

このように、現状の性能管理に関しては、管理者が関与する部分が少なくない。しかしながら、ヒューマンエラーの排除やその他管理コスト削減の観点から、できる限り管理は自動化することが好ましい。

【0011】

本発明は、上記従来技術の問題点を解決するためになされたもので、その目的は、DBMSが稼働する計算機システムにおいて、記憶装置に対する性能管理を、ユーザの業務における性能指標を用いておこなえるようにし、性能管理を容易化することである。

【 0 0 1 2 】

また、本発明のさらなる目的は、DBSMが稼動する計算機システムにおいて、記憶装置の性能チューニング処理を自動化し、性能管理コストを削減することである。

【 0 0 1 3 】

【課題を解決するための手段】

本発明の目的を達成するために以下のことを実施する。

【 0 0 1 4 】

先ず、計算機システムの構成要素の動作状態を把握するために、各構成要素においてそれらの稼動状況や、業務の応答時間等をモニタし、それを管理サーバに集約する。管理サーバは、収集したモニタ情報や要求性能等の予め与えられた情報を用いて、処理を実施するときや、モニタ値からチューニングが必要と判断した場合には、記憶装置へ設定変更指示を出す。設定変更内容としては、記憶装置のポートにおける処理量の割り当てや、データに対するキャッシュ量の割り当て、データを記憶するディスク装置の記憶領域の構成変更等がある。

【 0 0 1 5 】

バッチ業務をおこなうJob（管理サーバにおける処理を実現するプログラムの管理単位）に対しては、処理時間を見積る方法を管理サーバに与え、これを用いて処理時間を見積る。この見積りを用いて、要求時間内に処理が終わるように記憶装置の設定を変更する。処理時間は、データ処理量と現在利用可能な記憶装置のI/O処理量を考慮して見積る。処理時間を見積る方法は、実際に処理を実施させて計算機システムの各構成要素の動作状況を確認し、それと処理内容の対応付けをすることにより定めることができる。

【 0 0 1 6 】

オンライン業務をおこなうJobに関しては、処理の応答時間やスループットを計測し、要求性能に満たない場合に記憶装置の設定を変更する指示を出す。記憶装置のリソースの稼働率を参照して性能ボトルネックとなっている部分を発見し、それを解消するように設定変更指示を発行する。また、必要に応じて、アクセス性能を向上させるためにキャッシュ割り当ての変更指示を発行する。

## 【 0 0 1 7 】

計算機システム内で複数の J o b が並列して実施される場合には、異なる J o b 間で記憶装置内のリソースが共有され、性能ボトルネックになる可能性がある。そこで、J o b 毎に処理優先度を設け、リソース共有により問題が発生した場合には、優先度の低い J o b が利用している処理量の一部を優先度が高い J o b に割り当てることにより、優先度の高い J o b の要求性能を維持する制御をおこなう。

## 【 0 0 1 8 】

## 【発明の実施の形態】

以下、本発明に係る各実施形態を、図 1 ないし図 2 8 を用いて説明する。

## 【 0 0 1 9 】

## 〔実施形態 1〕

以下、本発明に係る第一の実施形態を、図 1 ないし図 2 7 を用いて説明する。

## (I) 本発明を適用する計算機システムの構成

先ず、図 1 を用いて本発明の第一の実施形態に係る記憶装置の性能を管理する計算機システムの構成について説明する。

図 1 は、本発明の第一の実施形態に係る記憶装置の性能を管理する計算機システムの構成図である。

## 【 0 0 2 0 】

計算機システムは、記憶装置 4 0、記憶装置 4 0 を使用する計算機（以下「サーバ」） 7 0、システム性能その他の管理をおこなう計算機（以下「管理サーバ」） 1 2 0、および、記憶領域の仮想化処理をおこなう仮想化スイッチ 6 0 を有する。各々の装置は、ネットワーク I / F 2 2 を有し、それを介してネットワーク 2 4 に接続され、相互に通信可能である。

## 【 0 0 2 1 】

サーバ 7 0、仮想化スイッチ 6 0、および、記憶装置 4 0 は、各々 I / O パス I / F 3 2 を有し、それを介して通信線（以下「I / O パス」） 3 4 により接続される。サーバ 7 0 と記憶装置 4 0 間の I / O 処理は、I / O パス 3 4 を用いて行われる。なお、I / O パス 3 4 は、装置間で異なる物理媒体や異なるプロトコ

ルでデータ転送をおこなう通信線が用いられてもよい。また、ネットワーク 2 4 と I / O パス 3 4 が同一の通信線でもよい。

【 0 0 2 2 】

記憶装置 4 0 は、CPU 1 2、メモリ 1 4、ディスク装置（以下「HDD」（Hard Disk Drive））1 6、ネットワーク I / F 2 2、I / O パス I / F 3 2 を有し、それらは、内部バス 1 8 で接続される。なお、HDD 1 6 は単数でも複数でもよい。メモリ 1 4 は、不揮発領域（ROM 領域）と高性能領域（RAM 領域）を有する。

【 0 0 2 3 】

本実施形態では、記憶装置は論理ディスク装置（以下、「LU」（Logical Unit））を外部装置に対して提供し、データのアクセス単位としてブロックを用いることにする。

【 0 0 2 4 】

記憶装置 4 0 を制御するプログラムである制御プログラム 4 4 は、メモリ 1 4 の不揮発領域に記憶され、起動時にメモリ 1 4 の高性能領域へ移された後に CPU 1 2 により実行される。記憶装置 4 0 が有する機能は、全て制御プログラム 4 4 により実現・制御される。

【 0 0 2 5 】

また、メモリ 1 4 には制御プログラム 4 4 が記憶装置 4 0 を制御・管理するために利用する管理情報 4 6 が記憶される。さらに、メモリ 1 4 の一部は外部装置からアクセス要求のあったデータを一時的に記憶しておく領域であるデータキャッシュ 4 2 に割り当てられる。

【 0 0 2 6 】

記憶装置 4 0 は、HDD 1 6 が有する物理記憶領域を仮想化して、一つまたは複数の LU 2 0 8 を外部装置に対して提供する。LU 2 0 8 は、HDD 1 6 と一対一に対応してもよいし、複数の HDD 1 6 から構成される記憶領域と対応してもよい。また、一つの HDD 1 6 が複数の LU 2 0 8 に対応してもよい。その対応関係は、管理情報 4 6 中に領域マッピング情報 3 0 0 の形で保持される。記憶装置 4 0 は、LU 2 0 8 の作成・削除、LU 2 0 8 の記憶領域の拡大・縮小を動

的に（従来技術の項の定義参照）おこなう機能を有する。また、記憶装置40は、LU208とそのデータを記憶するHDD16の記憶領域との対応関係をデータ移動により動的に変更する機能を有する。

## 【0027】

記憶装置40は、LU208をアクセス可能なI/OパスI/F32のI/Oパス34へのポート26を一つ以上有する。また、ネットワークI/F22におけるネットワーク24へのポートを有する。

## 【0028】

ポートとアクセスするデータの対応関係は、上位ポートマッピング情報322（図4で後述）の形で管理情報46中に保持される。また、記憶装置40は、LU208をアクセス可能なポート26を動的に追加・削除する機能を有する。

## 【0029】

記憶装置40では、LU208を単位として記憶領域のグループ化が行われ、そのグループ毎に独立したデータキャッシュ42内の領域が割り当てられる（以下、このLU208のグループを「キャッシュグループ」と称する）。キャッシュグループの作成・削除や、それに所属するLU208の追加・削除は、動的におこなうことができる。また、記憶装置40は、キャッシュグループへのデータキャッシュ42の割当量を動的に変更する機能を有する。キャッシュグループにおけるキャッシュデータ置換制御は、それぞれ独立にLRU置換アルゴリズムを用いておこなう。また、前記アルゴリズムによるデータ置換によりデータキャッシュ42に存在しなくなったデータの記憶領域に関しても、置換後も一定量分は仮想管理領域としてLRU置換アルゴリズム用の管理リスト中にその情報を記憶し、稼動状況の計測に利用する。

## 【0030】

記憶装置40は、I/OパスI/F32のポート26毎に、特定の外部装置からアクセス要求を、単位時間あたりの処理量が設定値を満たすように優先的に実行する機能（以下「優先アクセス制御機能」）を有する。その設定は、管理情報46中にポートスループット設定情報440（図8で後述）の形で保持される。記憶装置40は、ポート26における優先アクセス制御機能の設定を作成・変更

- ・削除する機能を有する。

## 【0031】

制御プログラム44は、記憶装置40の現存する各部位の稼動状況を計測し、それを管理情報46中にストレージモニタ情報360の形で記憶する。特に、キャッシュヒット数統計に関し、LRU置換アルゴリズム用のキャッシュセグメント（データキャッシュ42の領域管理単位）の管理リストを等サイズの複数領域に分割し、分割領域毎にヒット数を計測する。さらに、仮想管理領域に関しても実管理領域（実際にデータに対してキャッシュセグメントが割り当てられている領域）と等サイズの領域に分割して実管理領域と同様にヒット数を計測する。

## 【0032】

記憶装置40は、外部からの要求に応じて、領域マッピング情報300やストレージモニタ情報360、ポートスループット設定情報440、その他記憶装置40の構成情報を、ネットワーク24を介して外部に送信する機能を有する。なお、ここに出てきたデータ構造については、いずれも詳細に後述する。

## 【0033】

また、外部装置からネットワーク24を介して外部から受信した指示に従い、前述の各種機能を実行する機能を有する。

## 【0034】

仮想化スイッチ60は、CPU12、メモリ14、ネットワークI/F22、I/OバスI/F32を有し、それらは内部バス18で接続される。メモリ14は、不揮発領域（ROM領域）と高性能領域（RAM領域）を有する。

## 【0035】

仮想化スイッチ60を制御するプログラムである制御プログラム64は、メモリ14の不揮発領域に記憶され、起動時にメモリ14の高性能領域へ移された後にCPU12により実行される。仮想化スイッチ60が提供する機能は、全て制御プログラム64により制御される。また、メモリ14には、制御プログラム64が仮想化スイッチ60を制御・管理するために利用する管理情報66が記憶される。

## 【0036】



仮想化スイッチ60は、本装置に接続された記憶装置40から提供されるLU208を認識し、その記憶領域を仮想化して仮想ボリューム206を外部装置に提供する。なお、仮想化スイッチ60が多段接続された場合には、他の仮想スイッチ60が提供する仮想ボリューム206を記憶装置40から提供されるLU208と等価に扱い、その記憶領域を仮想化して仮想ボリューム206を外部装置に提供する。その対応関係は管理情報66中に領域マッピング情報300の形で保持される。仮想化スイッチ60は、仮想ボリューム206の作成・削除、仮想ボリューム206の記憶領域の拡大・縮小を動的におこなう機能を有する。さらに、仮想ボリューム206を構成するLU208や他の仮想ボリューム206の記憶領域との対応関係を動的に変更する機能を有する。また、仮想化スイッチ60は、それが保持する領域マッピング情報300で後述するフリーエントリ314に対応する領域について、領域マッピング情報300から削除する（以下「領域を開放する」）機能を有する。

#### 【0037】

仮想化スイッチ60では、仮想ボリューム206をアクセス可能なI/OパスI/F32のポート26を一つ以上有する。その対応関係は、上位ポートマッピング情報322（後述）の形で管理情報66中に保持される。また、仮想ボリューム206をアクセス可能なポート26を動的に追加・削除する機能を有する。

#### 【0038】

仮想化スイッチ60が提供する仮想ボリューム206がアクセスされた際、実施されたアクセス回数やその種別、転送されたデータ量をI/OパスI/F32のポート26毎に計測し、それを管理情報66中にポートモニタ情報400（図9で後述）の形で記憶する。

#### 【0039】

仮想化スイッチ60は、外部からの要求に応じて、領域マッピング情報300やポートモニタ情報360を、ネットワーク24を介して外部に送信する機能を有する。また、外部装置からネットワーク24を介して外部から受信した指示に従い、前述の各種機能を実行する機能を有する。なお、これらのデータ構造については、後に詳細に説明する。

## 【0040】

サーバ70は、CPU12、メモリ14、HDD16、ネットワークI/F22、I/OバスI/F32を有し、それらは内部バス18で接続される。

## 【0041】

メモリ14上には、オペレーティングシステム(OS, Operating System)72と管理エージェント144がHDD16から読み込まれ、CPU12により実行される。

## 【0042】

OS72には、デバイスドライバ76、ボリュームマネージャ78、ファイルシステム80が含まれる。メモリ14上に読み込まれたOS72は、それらや他のOS72を構成するプログラムが利用する管理情報であるOS管理情報74を有する。OS管理情報74には、CPU12の数等、サーバ70のハードウェア構成の情報に含まれる。また、OS72は、OS管理情報74中に記憶されている情報を外部プログラムが読むためのソフトウェアインターフェイスを有する。なお、図1に示した例では、サーバ70は一つのファイルシステム80しか有していないが、複数のファイルシステム80を有してもよい。OS72は、OS72が動作している計算機で実施されるプログラムの管理をおこなう。このとき、実行中のプログラム全体がCPU12をどれだけ利用したかを計測、それを基にCPU12の稼働率を計算し、OS管理情報74中に保持する。

## 【0043】

デバイスドライバ76は、ネットワークI/F22やI/OバスI/F32等のハードウェアを制御するプログラムである。ネットワークI/F22やI/OバスI/F32を通してデータ転送(アクセス)をおこなった際に、実施された回数やその種別、転送されたデータ量をネットワークI/F22やI/OバスI/F32のポート26毎に計測し、それをOS管理情報74中にポートモニタ情報400の形で記憶する。

## 【0044】

ボリュームマネージャ78は、記憶装置40から提供されるLU208や仮想化スイッチ60から提供される仮想ボリューム206の記憶領域を仮想化し、論

理ボリューム204をファイルシステム80に提供するプログラムである。その対応関係はOS管理情報74中に領域マッピング情報300の形で保持される。また、ボリュームマネージャ78は論理ボリューム204の構成変更（作成・削除を含む）を動的に実施する機能を有し、ボリュームマネージャ78が有するソフトウェアインターフェイスにより実行指示を与えることができる。ボリュームマネージャ78は、それが保持する領域マッピング情報300でフリーエントリ314に対応する領域を開放する機能を有する。また、ボリュームマネージャ78は、複数I/Oパス34を利用したI/O処理の負荷分散機能を有してもよい。

#### 【0045】

ファイルシステム80は、記憶装置40から提供されるLU208や仮想化スイッチ60から提供される仮想ボリューム206、ボリュームマネージャ78から提供される論理ボリューム204の記憶領域を仮想化し、ファイル202を他のプログラムに提供するプログラムである。その対応関係はOS管理情報74中に領域マッピング情報300の形で保持される。なお、ファイル202と同じソフトウェアインターフェイスで、論理ボリューム204、仮想ボリューム206、LU208の記憶領域に直接アクセスする機能であるローデバース機能も、ファイルシステム80により提供される。

#### 【0046】

管理エージェント144は、管理サーバ120上で実行されるシステム管理プログラム140からネットワーク24を介して受け付けた処理要求をサーバ70で実行し、必要に応じてその結果をネットワーク24を介してシステム管理プログラム140に返信するプログラムである。管理エージェント144が実施する処理は、1) OS管理情報74中に記憶されている情報の読み出し、2) DBMS90やAPプログラム100の実行・停止、3) DBMS管理情報92中に記憶されている情報の読み出し、4) DBMS90にSQL (Structured Query Language) 文を与え、SQL実行計画を取得、5) APプログラム管理情報102中に記憶されている情報の読み出し、6) ボリュームマネージャ78に対する構成変更指示、7) OS72に対する各種設定指示、等である。

## 【0047】

DBMS90は、DBに関する一連の処理・管理を実行するサーバ70上で実行されるプログラムである。DBMS90は、システム管理プログラム140からの指示に従って管理エージェント144から起動されて、サーバ70内のHDD16または記憶装置40からメモリ14に読み出されてCPU12により実行される。

## 【0048】

DBMS90は、記憶領域の管理単位として、一つ以上のファイル202が提供する記憶領域からなる「データ領域」を用いる。DBMS90が利用・管理する表・索引・ログ等（以下、まとめて「データ構造」と称する）は、それぞれ記憶されるべきデータ領域が定められている。データ領域の一部は、データ構造に対して未割り当てでもよい。

## 【0049】

メモリ14上に読み込まれたDBMS90は、データ記憶領域情報342を含むDBMS90の管理情報であるDBMS管理情報92を有する。DBMS90は、DBMS管理情報92を外部プログラムが読むためのソフトウェアインターフェイスを有する。

## 【0050】

DBMS90は、SQL文を与えるとその実行計画（以下、「SQL実行計画」と称す）を作成したのちにそれに従って処理を実行する。基本的に、検索条件以外同じSQL文を与えると同じSQL実行計画を作成する。DBMS90は、与えたSQL文のSQL実行計画を出力するソフトウェアインターフェイスを有する。また、SQL実行計画を出力するときに、DBMS90が見積もった処理件数等を同時に出力してもよい。

## 【0051】

APプログラム100は、ユーザがおこなう業務のためにサーバ70上で実行されるプログラムであり、DBMS90に対して処理要求を発行するものとする。APプログラム100は、システム管理プログラム140からの指示に従って管理エージェント144から起動され、サーバ70内のHDD16または記憶装

置 40 からメモリ 14 に読み出されて CPU 12 により実行される。メモリ 14 上に読み込まれた AP プログラム 100 は、その管理情報である AP プログラム管理情報 102 を有する。なお、AP プログラム 100 は記憶装置 40 に記憶されるデータを扱う際に常に DBMS 90 に処理要求を発行してもよく、その場合には、AP プログラム 100 が実行されるサーバ 70 は I/O パス I/F 32 を有さなくてもよい。

## 【0052】

本実施形態におけるオンライン業務のために実行される AP プログラム 100 は、一つ以上の処理の組として実現され、処理それぞれに識別子である処理 ID 432 がつけられる。ユーザはそのいずれかの処理実行を要求し、AP プログラム 100 はそれに応じて処理を実施する。AP プログラム 100 は受け付けた処理要求のキューイング制御を実施し、AP プログラム 100 から DBMS 90 に対して処理要求を発行した場合には、DBMS 90 で即座に処理を開始できるものとする。AP プログラム 100 は、処理に関する実行統計情報を取得し、それをオンライン Job モニタ情報 430 (図 10 で後述) として AP プログラム管理情報 102 中に保持する。このとき、AP プログラム 100 は、AP プログラム管理情報 102 を外部プログラムが読むためのソフトウェアインターフェイスを有する。

## 【0053】

DBMS 90 や AP プログラム 100 は、1 台のサーバ 70 上で複数同時に実行することができる。また、DBMS 90 と AP プログラム 100 が異なるサーバ 70 上で実行されてもよく、その場合には、AP プログラム 100 は DBMS 90 に処理要求をネットワークを経由して伝達する。

## 【0054】

OS 72 や DBMS 90、AP プログラム 100、管理エージェント 144 は、それらを記憶した CD-ROM (記憶媒体) が管理サーバ 120 が有する CD-ROM ドライブ 20 を用いて読み出され、ネットワーク 24 を介してサーバ 70 内の HDD 16 もしくは記憶装置 40 にインストールされる。

## 【0055】

管理サーバ120は、CPU12、メモリ14、HDD16、CD-ROMドライブ20、ネットワークI/F22を有し、それらは内部バス18で接続される。

【0056】

メモリ14上には、OS72とシステム管理プログラム142がHDD16から読み込まれ、CPU12により実行される。CD-ROMドライブ20は、各種プログラムのインストールに用いられる。

【0057】

また、管理サーバ120には、キーボード・マウス等の入力装置112および表示画面114を有する管理端末110がネットワーク24を介して接続される。この接続は、ネットワーク24とは異なる通信線を用いてもよく、管理サーバ72と管理端末110が一体で構成されてもよい。管理者は、原則、情報の入出力を管理端末110を介して行い、必要に応じてCD-ROMドライブ20等も用いる。

【0058】

システム管理プログラム140は、管理サーバ120が有する、性能その他のシステム管理機能を実現するプログラムである。HDD16からメモリ14上に読み込まれ、CPU12により実行される。システム管理プログラム140は、その機能を実現するために必要な管理情報であるシステム管理情報142を有する。本プログラムは、それを記憶したCD-ROMが管理サーバ120が有するCD-ROMドライブ20を用いて読み出され、HDD16にインストールされる。本プログラムの動作詳細については後述する。

【0059】

なお、本実施形態の例では、システム管理プログラム140は管理サーバ120上で実行されるものとして説明するが、任意のサーバ70、仮想化スイッチ60、記憶装置40上で実行されてもよい。このとき、サーバ70では、HDD16に記憶され、メモリ14に読み出された後にCPU12により実行される。仮想化スイッチ60、記憶装置40では、メモリ14の不揮発領域に記憶され、メモリ14の高性能領域に移された後にCPU12により実行される。

## (II) データマッピングの階層構造

次に、図 2 を用いて本実施形態の DBMS 9 0 が管理するデータのデータマッピングの階層構成について説明する。

図 2 は、本発明の第一の実施形態に係る DBMS 9 0 が管理するデータのデータマッピングの階層構成図である。

## 【 0 0 6 0 】

本実施形態の例では、サーバ 7 0 と記憶装置 4 0 との間に一つの仮想化スイッチ 6 0 が存在する場合を説明する。なお、以下、ある 2 つの階層について、DBMS 9 0 に近い方（すなわち、アプリケーション側）を「上位」、HDD 1 6 に近い方（すなわち、物理的記憶手段側）を「下位」の階層と称する。また、記憶装置 4 0、仮想化スイッチ 6 0、ボリュームマネージャ 7 8、ファイルシステム 8 0 をまとめて「仮想化機構」と称する。ファイル 2 0 2、論理ボリューム 2 0 4、仮想ボリューム 2 0 6、LU 2 0 8 をまとめて「仮想構造」と称し、さらに、仮想構造に HDD 1 6 を加えたものをまとめて「管理構造」と称する。

## 【 0 0 6 1 】

図 2 で示した例では、DBMS 9 0 は、それが管理しているデータ構造 2 0 0 を記憶しているファイルシステム 8 0 が提供するファイル 2 0 2 に対してアクセスをおこなう。ファイルシステム 8 0 は、ファイル 2 0 2 に対するアクセスを対応する論理ボリューム 2 0 4 の領域へのアクセスに変換する。ボリュームマネージャ 7 8 は、論理ボリューム 2 0 4 に対するアクセスを対応する仮想ボリューム 2 0 6 の領域へのアクセスに変換する。仮想化スイッチ 6 0 は、仮想ボリューム 2 0 6 に対するアクセスを対応する LU 2 0 8 の領域へのアクセスに変換する。記憶装置 4 0 は、LU 2 0 8 に対するアクセスを、対応する HDD 1 6 に対するアクセスに変換する。このように、仮想化機構は、それが上位階層に提供する仮想構造のデータを下位階層に存在する一つ以上の管理構造の記憶領域にマッピングする。

## 【 0 0 6 2 】

また、図示しないが、ある仮想構造のデータの同一部分が複数の下位階層の管理構造にマッピングされてもよい。また、ある仮想構造のデータが HDD 1 6 に

マッピングされる経路が複数存在してもよい。これらの場合には、仮想化機構が、そのようなマッピングであることを領域マッピング情報 300 中に保持しておく。さらに、ある管理構造が複数のサーバ 70 に共有されるマッピングを有してもよい。

#### 【0063】

本実施形態では、論理層 212 における管理構造間のデータの対応関係が明確化されればよく、サーバ 70 でボリュームマネージャ 78 が使用されなくてもよい。さらに、仮想化スイッチ 60 は、複数存在してもよいし、仮想化スイッチ 60 が存在せずにサーバ 70 と記憶装置 40 が I/O パス 34 により直結されてもよい。仮想化スイッチ 60 に相当するスイッチが記憶領域の仮想化機能を有しなくてもよく、この場合、仮想化スイッチ 60 が下位階層から提供された管理構造をそのまま上位階層へ仮想構造として提供していると考ええる。

#### 【0064】

なお、本実施形態の階層構成のモデルで、一つの LU 208 には一つのデータ構造 208 のみが記憶されるようにマッピングを作成すると、後述する記憶装置 40 の構成変更を変更した際の効果見積の精度向上等に役立つというメリットがある。

#### (III) データ構造

次に、図 3 ないし図 17 を用いて本実施形態の性能管理方法に用いられるデータ構造について説明する。

#### 【0065】

まず、図 3 を用いて領域マッピング情報 300 を説明する。

図 3 は、領域マッピング情報 300 のデータ構造を示す模式図である。

#### 【0066】

領域マッピング情報 300 は、各仮想化機構の管理情報の中に置かれ、仮想化機構が提供する仮想構造の領域と、それが利用する管理構造の領域の対応を保持するものであり、エントリ 302、304 を有する。

#### 【0067】

なお、ここでの説明では、上位仮想構造と下位管理構造を対応させるものとし



て、モデル化しているが、実際には、各対応する仮想化機構に応じて、（ファイル、論理ボリューム）、（論理ボリューム、仮想ボリューム）、（仮想ボリューム、論理ディスク装置）、（論理ディスク装置、HDD）の対応を示すものになる。

## 【0068】

エントリ302は、仮想化機構が上位階層に提供する仮想構造の領域に関する情報であり、仮想構造の識別子である仮想構造ID306を保持するエントリ、その構造内の領域を示すエントリ、その領域が下位階層の複数の仮想構造へ記憶されたり、異なる経路でHDD16に対応している場合に、それらを識別するための多重化方法を示すエントリの組を有する。エントリ304は、エントリ302に対応する下位階層の管理構造の領域に関する情報であり、管理構造をアクセスする際に利用するI/OパスI/F32のポート26の識別子であるポートID326を保持するエントリ、管理構造を提供する仮想化機構の識別子である仮想化機構ID308を保持するエントリ、管理構造の識別子である管理構造ID310を保持するエントリ、その構造内領域を示すエントリの組を有する。なお、ファイルシステム80がボリュームマネージャ78が提供する論理ボリューム204を利用する場合には、ポートID326を保持するエントリを保持しない。また、記憶装置40においては、ポートID326と仮想化機構ID308を有するエントリを保持しない。

## 【0069】

仮想構造ID306が“Free”であるフリーエントリ314は、本領域マッピング情報300を有する仮想化機構が、その領域を利用できる状態にあるが、まだ仮想構造として利用していない下位階層の記憶領域を示すことにする。データ移動による対応関係の動的変更機能を有する仮想化機構は、フリーエントリ314に対応する管理構造の記憶領域をデータ移動先として利用する。

## 【0070】

本実施形態においては、異なる仮想構造が同一の管理構造の記憶領域を利用することが許されるものとする。また、仮想化機構ID308、仮想構造ID306、管理構造ID310、ポートID326はシステム内で一意に定まる識別子

であるとする。あるいは、システム内で一意でない場合でも、装置の識別子を付加することによりシステム内で一意に定まるようにする。

【0071】

次に、図4を用いて上位ポートマッピング情報322について説明する。

図4は、上位ポートマッピング情報322のデータ構造を示す模式図である。

【0072】

上位ポートマッピング情報322は、仮想化スイッチ60の管理情報66および記憶装置40の管理情報46の中に置かれ、あるポートを通して外部装置に提供される仮想構造を管理する情報であり、ポートID326を保持するエントリと、対応する仮想構造ID306を保持するエントリの組を保持する。

【0073】

次に、図5を用いてデータ記憶領域情報342について説明する。

図5は、データ記憶領域情報342のデータ構造を示す模式図である。

【0074】

データ記憶領域情報342は、サーバ70のDBMS管理情報92の中に置かれ、DBMS90が管理するデータの記憶領域管理に用いる情報であり、データ領域の名前であるデータ領域名344を保持するエントリ、データ構造の名前であるデータ構造名346を保持するエントリ、対応データがファイル202のどの位置に記憶されているかの情報であるデータ記憶位置348を保持するエントリの組からなる。

【0075】

次に、図6を用いてキャッシュグループ情報460について説明する。

図6は、キャッシュグループ情報460のデータ構造を示す模式図である。

【0076】

キャッシュグループ情報460は、記憶装置40の管理情報46の中に置かれ、記憶装置40がキャッシュグループの管理に用いる情報であり、キャッシュグループの識別子であるキャッシュグループID462を保持するエントリと、そのキャッシュグループに対して割り当てられるキャッシュセグメント数の情報である割当セグメント数464を保持するエントリ、そのキャッシュグループに属

する L U 2 0 8 の識別子である L U I D 3 6 4 を保持するエントリの組を有する。

【 0 0 7 7 】

次に、図 7 を用いてストレージモニタ情報 3 6 0 について説明する。

図 7 は、ストレージモニタ情報 3 6 0 のデータ構造を示す模式図である。

【 0 0 7 8 】

ストレージモニタ情報 3 6 0 は、記憶装置 4 0 の管理情報 4 6 の中に置かれ、記憶装置 4 0 の稼動状況に関するデータを保持しており、キャッシュモニタ情報 3 6 2、ストレージポートモニタ情報 3 8 2、HDD モニタ情報 3 9 2 を含んでいる。

【 0 0 7 9 】

キャッシュモニタ情報 3 6 2 は、L U 2 0 8 が外部装置からアクセスされた場合にどのキャッシュヒットしたかに関する稼動統計情報である。

【 0 0 8 0 】

キャッシュモニタ情報 3 6 2 は、実管理領域の分割数を示す実領域分割数 5 0 2 と、実管理領域の分割後のサイズを単位とした仮想管理領域の保持量を示す仮想管理領域数 5 0 4 を含む。また、L U 2 0 8 毎のキャッシュヒット数の統計値として、L U I D 3 6 4 を保持するエントリと、それに対応するリード／ライトの I ／ O 種別 3 6 6 を保持するエントリ、それに対応する I ／ O 処理の累積実行回数 3 6 8 を保持するエントリ、そのときにヒットした回数を示すキャッシュヒット回数累計 3 7 0 を保持するエントリの組を保持する。なお、キャッシュヒット回数累計 3 7 0 は、実管理領域でのヒット数の総累計値と、実管理領域と仮想管理領域の双方で分割された各領域でのヒット数の累計値を保持する。本実施形態では、実管理領域と仮想管理領域を統合して分割された領域に通番を付加し、最も最近利用されたデータが存在する領域を第 1 領域、以下小さい通番の領域ほど最近利用されたデータを有する領域であるとする。

【 0 0 8 1 】

ストレージポートモニタ情報 3 8 2 は、L U 2 0 8 毎に I ／ O パス I ／ F 3 2 のポート 2 6 を利用してどれだけデータ転送処理が実施されたかの実行統計情報

で、ポート I D 3 8 4 を保持するエントリと L U I D 3 6 4 を保持するエントリ、それに対応する I / O 処理の累積実行回数 3 6 8 を有するエントリと累積データ転送量 3 8 6 を保持するエントリの組を有する。

## 【 0 0 8 2 】

HDD モニタ情報 3 9 2 は、L U 2 0 8 毎に HDD 1 6 をどれだけ利用しているかの実行統計情報で、HDD の識別子である HDD I D 3 9 4 を保持するエントリと L U I D 3 6 4 を保持するエントリ、それに対応する実際に処理をおこなっていた時間（待ち時間を含まない）の累積値である累積処理時間 3 9 6 を保持するエントリの組を保持する。

## 【 0 0 8 3 】

次に、図 8 を用いてポートスループット設定情報 4 4 0 について説明する。

図 8 は、ポートスループット設定情報 4 4 0 のデータ構造を示す模式図である。

## 【 0 0 8 4 】

ポートスループット設定情報 4 4 0 は、記憶装置 4 0 の管理情報 4 6 の中に置かれ、特定のポート 2 6 に対して外部装置の特定ポート 2 6 からアクセス処理量を制限する優先アクセス制御機能のための設定情報である。このポートスループット設定情報 4 4 0 は、ポート I D 3 8 4 を保持するエントリとアクセス元ポート I D 4 4 2 を保持するエントリ、最大データ転送量 4 4 4 を保持するエントリと、最大 I / O 処理量 4 4 4 を保持するエントリ 4 4 6 の組を有する。

## 【 0 0 8 5 】

アクセス元ポート I D 4 4 2 とは、制限をかけるポート 2 6 にアクセスしてくる外部装置のポート 2 6 のポート I D である。最大データ転送量 4 4 4 とは、ポート 2 6 に対応する単位時間あたりの最大転送量設定値である。最大 I / O 処理量 4 4 4 とは、単位時間あたりの最大 I / O 処理量の設定値である。

## 【 0 0 8 6 】

次に、図 9 を用いてポートモニタ情報 4 0 0 について説明する。

図 9 は、ポートモニタ情報 4 0 0 のデータ構造を示す模式図である。

## 【 0 0 8 7 】

ポートモニタ情報400は、仮想化スイッチ60の管理情報66およびサーバ70のOS管理情報74の中に置かれ、ポート26によりおこなわれたデータ転送処理に関する実行統計情報であり、ポートID384を保持するエントリと、対応するI/O種別366を保持するエントリ、I/O処理の累積実行回数368を有するエントリ、処理要求の発行/受取から処理完了までの時間（待ち時間を含む）である応答時間の累積値である累積応答時間402を保持するエントリ、累積データ転送量386を保持するエントリの組を有する。

## 【0088】

次に、図10を用いてオンラインJobモニタ情報430について説明する。

図10は、オンラインJobモニタ情報430のデータ構造を示す模式図である。

## 【0089】

オンラインJobモニタ情報430は、サーバ70のAP管理情報102の中に置かれ、APプログラム100が取得するオンラインJobのための実行統計情報であり、処理ID432を保持するエントリ、対応する処理の累積実行回数368を保持するエントリ、その処理の累積応答時間402を保持するエントリ、その処理をDBMS90に対して発行した際の累積処理時間396を保持するエントリの組を有する。

## 【0090】

次に、図11を用いてストレージ性能情報610について説明する。

図11は、ストレージ性能情報610のデータ構造を示す模式図である。

## 【0091】

ストレージ性能情報610は、管理装置120のシステム管理情報142中に保持される情報であり、HDD性能情報612とポート性能情報616を有する。この情報は、記憶装置40やその対応部位の型式を取得し、それと事前に与えられた型式毎の性能情報と組み合わせて作成される。

## 【0092】

HDD性能情報612は、HDD16のアクセス性能に関する情報であり、記憶装置40の識別子である装置ID572を保持するエントリとHDDID39

4 を保持するエントリ、それに対応する HDD 16 が有するアクセス性能に関する情報であるアクセス性能情報 612 を含むエントリの組を有する。

【0093】

アクセス性能情報 612 は、ランダムアクセス時のリードでキャッシュヒットした／しない（ミスした）場合、ライトでキャッシュヒットした／しない場合のそれぞれの平均応答時間を保持する。

【0094】

ポート性能情報 616 は、記憶装置 40 の I/O パス I/F 32 のポート 26 の最大性能に関する情報であり、装置 ID 572 を保持するエントリとポート ID 384 を保持するエントリ、対応するポート性能情報 618 を含むエントリの組を有する。ポート性能情報 612 は、そのポートの最大性能の情報として最大 I/O 処理量（IOPS）と最大データ転送量（MB/s）の組を保持する。

【0095】

次に、図 12 を用いてモニタ履歴情報 510 について説明する。

図 12 は、モニタ履歴情報 510 のデータ構造を示す模式図である。

【0096】

モニタ履歴情報 510 は、管理装置 120 のシステム管理情報 142 中に保持される情報であり、各装置やプログラムが集めるモニタ情報であり、これまで説明してきたキャッシュモニタ情報 370、ストレージポートモニタ情報 382、HDD モニタ情報 392、ポートモニタ情報 400、オンライン Job モニタ情報 430、そして、OS 72 が計算する CPU 12 の稼働率（以下これらをまとめて「モニタ情報」と称す）を取得し、それを適当な形に編集した情報を履歴情報として保持するものである。

【0097】

稼働状況がモニタされている部位（キャッシュモニタ情報 370、ストレージポートモニタ情報 382、HDD モニタ情報 392 に関しては、アクセス先の LU 208 も含む）に関して、システム内で一意に定まるモニタ部位 ID 514 が付加される。

【0098】

モニタ履歴情報 510 は、モニタ部位 ID 514 を保持するエントリとそのモニタ内容 516 を示す情報を保持するエントリと、対応する前回取得情報 518 を保持するエントリと、履歴情報 520 を保持するエントリの組を有する。モニタ内容 516 の具体的な内容としては、各モニタ情報のデータ構造を説明する際に説明した通りである。

## 【0099】

前回取得情報 518 は、前回モニタ情報の値を取得した日時のエントリとその取得値のエントリを有する。

## 【0100】

履歴情報 520 は、履歴として集計・記憶される値の内容を示す履歴内容 522 を保持するエントリと、ある期間のモニタ値であるサンプル 524 を保持する複数のエントリからなる。本実施形態における履歴内容 522 としては、ストレージポートモニタ情報 382 から記憶装置 40 の I/O パス I/F 32 のポート 26 における LU208 毎の平均実行回数と平均データ転送量、キャッシュモニタ情報 370 から LU208 におけるリード/ライト毎の平均実行回数と平均実ヒット率と、リードの領域毎の平均ヒット率（期間内の領域でのリードヒット回数/期間内の LU208 へのリード実行回数）、HDD モニタ情報から HDD 16 における LU208 毎の平均稼働率、ポートモニタ情報 400 から、サーバ 70、仮想化スイッチ 60 の I/O パス I/F 32 のポート 26 もしくはサーバ 70 のネットワーク I/F のポート 26 におけるリード/ライト毎の平均実行回数、平均データ転送量、平均応答時間、オンライン Job モニタ情報 430 からとして AP プログラム 100 における各処理毎の平均実行回数、平均応答時間、平均処理時間、OS 72 が計算する CPU 12 の稼働率からとしてサーバ 70 における CPU 稼働率がある。

## 【0101】

サンプル 524 は、そこに記憶されたデータがいつの期間のモニタ値かを示すエントリと、その期間における集計値である平均値と最大値をそれぞれ保持するエントリからなる。

## 【0102】

次に、図13を用いてバッチJob管理情報700について説明する。

図13は、バッチJob管理情報700のデータ構造を示す模式図である。

【0103】

バッチJob管理情報700は、管理サーバ120のシステム管理情報142の中に置かれるバッチJobのための管理情報であり、Jobの識別子であるJobID622を保持するエントリ、Jobの実行優先度を示すJob優先度704を保持するエントリ、Job種別706を保持するエントリ、このJobがどのような開始条件で実行されるかを示す実行条件708を保持するエントリ、Jobの最大実行時間の設定値である最大実行時間710を保持するエントリ、Jobを実行するサーバ70の識別子である実行サーバID712を保持するエントリとそのサーバ70で実行するコマンドである実行コマンド714を保持するエントリの組、Job依存入力情報720、Job依存出力データ情報730、CPU処理時間見積方法716、Jobが利用するデータに関する情報であるI/O処理内容情報740aを含む。実行サーバID712と実行コマンド714の組に関しては、複数存在しても良い。

【0104】

Job種別706は、バッチJob管理情報700の場合には、バッチ業務向けであることを示す情報が保持される。実行条件708の内容として、バッチJob管理情報700では、実行すべき日付や曜日の条件を含めた実行開始時刻、あるいは、一つ以上の別のJobが完了した時に実行する等がある。

【0105】

本実施形態のJobの実行イメージとしては、バッチJobは連続して実行され、前段のJobの出力データを次段のJobの入力データとして引き継いでいくものを考える。

【0106】

Job依存入力情報720は、そのJobを実行する際に利用するデータに関する情報であり、利用するデータを出力するJobのJobID622である前段JobID722を保持するエントリと、その入力データの情報であるデータ情報724を保持するエントリと、データの行数やファイルサイズ等、データ量



の計量単位の情報であるデータ量単位726を保持するエントリの組からなる。

【0107】

データ情報724に記憶される情報は、基本的に、ファイル202の識別子またはDBMS90とそれが管理するデータ構造200の識別子の組等、記憶装置40に記憶されるデータの識別子であるが、制御情報受け渡しのための識別子でもよい。なお、入力に依存する処理をおこなわない場合、Job依存入力情報720は存在しない。

【0108】

Job依存出力データ情報730は、他のJob実行に利用されるJobの出力データに関する情報であり、出力データを利用するJobのJobID622である次段JobID732を保持するエントリと、その出力データのデータ情報724を保持するエントリ、データ量単位726を保持するエントリの組からなる。Job終了時に処理完了情報として返す出力データ量は、Job依存出力データ情報730中のデータ量単位726を用いて計測する。なお、出力データを利用するJobが存在しない場合には、Job依存出力データ情報730は存在しない。

【0109】

CPU処理時間見積方法716は、処理の実行時間のうちサーバ70のCPU12がボトルネックとなっている時間を見積る方法に関する情報である。

【0110】

I/O処理内容情報740aは、Jobが利用するデータに関する情報であり、このJobが利用するデータのデータ情報724を保持するエントリと、対応するデータの、アクセス順742を保持するエントリ、I/Oネック744を保持するエントリ、I/Oタイプ746を保持するエントリ、I/Oベース性能748を保持するエントリ、入力データ量から処理量を見積る方法の情報である処理量見積752を保持するエントリ、Job実行時に記憶装置40に対して指示すべき内容であるストレージヒント754を保持するエントリの組を有する。なお、ストレージヒント754は、記憶装置40に関する様々な内容が考えられるが、本実施形態では、データに対するキャッシュ量の明示的設定とした。

## 【0111】

アクセス順742はアクセス順に関する情報で、番号が若いほど先にアクセスされ、同じ値のものは同時にアクセスされることを意味する。I/Oネック744は、そのI/O処理が性能ボトルネックに、なる／ならないを示す情報であり、“Y” “N”の値を取る。I/Oタイプ746はI/O処理種別に関する情報であり、“シーケンシャル” “ランダム”の値を取る。I/Oベース性能748は、実行時間の見積り等を利用するI/O性能に関する情報であり、データの複数の記憶装置40への分散配置等に対応するため、そのデータをアクセスする際に利用される各記憶装置40の各ポート26毎にポートID384と性能値の組を保持する。性能値は、I/Oタイプ746が“シーケンシャル”の場合にはシーケンシャル性能（データ転送速度）、“ランダム”の場合にはランダム性能（IOPS）となる。また、処理量見積752はI/Oベース性能748に保持されるポートID384毎にその情報を保持する。

## 【0112】

次に、図14を用いてバッチSQL設計情報850を説明する。

図14は、バッチSQL設計情報850のデータ構造を示す模式図である。

## 【0113】

バッチSQL設計情報850は、バッチジョブ管理情報700を設定するために管理サーバ120に入力され、APプログラム100がDBMS90に対して要求する処理に関する情報である。このバッチSQL設計情報850は、設定をおこなうJobのJobID622を保持するエントリ、処理をおこなうDBMS90の識別子である実行DBMSID852を保持するエントリ、主入力データ854を保持するエントリ、SQLID858aを保持するエントリと対応する実行SQL文860を保持するエントリの組、データ情報724を保持するエントリとそれに対応するSQLID858bを保持するエントリの組、繰り返し実行SQL情報862を有する。

## 【0114】

主入力データ854とは、その量がAPプログラム100のCPU処理時間に一番大きく影響を与えるデータである。

## 【0115】

SQLID858aは、APプログラム100が実施する順に通番が振られ、APプログラム100がDBMS90に発行するSQL文の情報として、SQLID858と実行されるSQL文である実行SQL文860が対応して保持される。

## 【0116】

SQLID858bは、入出力データとしてファイル202を利用する場合に、そのデータ情報724と、そのデータ情報を利用したり、結果を保存するSQL文のIDである。

## 【0117】

繰り返し実行SQL情報862は繰り返し実行されるSQL文に関する情報であり、SQLIDセット864を保持するエントリと、その繰り返しの際の入力データのデータ情報724を保持するエントリの組を有する。SQLIDセット864とは、繰り返し実行される一つまたは複数のSQL文のSQLID858の組である。

次に、図15を用いてオンラインJob管理情報770について説明する。

図15は、オンラインJob管理情報770のデータ構造を示す模式図である。

## 【0118】

オンラインJob管理情報770は、管理サーバ120のシステム管理情報142の中に置かれ、オンラインJobを管理するための情報であり、JobID622を保持するエントリ、Job優先度704を保持するエントリ、Job種別706を保持するエントリ、実行条件708を保持するエントリ、要求性能774を保持するエントリ、実行サーバID712を保持するエントリと実行コマンド714を保持するエントリの組、I/O処理内容情報740c、処理の多重度を示す処理多重度782を保持するエントリを含む。

## 【0119】

本実施形態で、図13に示したバッチJob管理情報700とオンラインJob管理情報770とを分けたのは、そのJobの性質により要求される性能、管

理すべき情報が異なっていることを考慮したものである。

#### 【0120】

オンラインJob管理情報770の場合には、Job種別706はオンライン業務向けであることを示す情報が保持される。実行サーバID712と実行コマンド714の組に関しては、複数存在しても良い。実行条件708の内容として、オンラインJob管理情報770では、実行すべき日付や曜日の条件を含めた実行開始時刻と実行終了時刻（常時実行を含む）が与えられる。要求性能774はJobへの要求性能の設定情報であり、処理全体での平均応答時間と、高負荷状態における処理スループット（処理の平均実行回数）が保持される。

#### 【0121】

I/O処理内容情報740cは、内容的には図13に示したバッチJob管理情報700中のI/O処理内容情報740aの一部を抜き出したものであり、データ情報724を保持するエントリと、対応するI/Oベース性能748を保持するエントリとストレージヒント754を保持するエントリの組を有する。Jobの設計情報から処理がアクセスするデータを把握し、データ情報724を設定し、I/Oベース性能748のポートID386をマッピング集約情報を参照して求める。ストレージヒント754に関しては、記憶装置に対する指示情報を設定する。本実施形態では、必要に応じてデータ毎に利用したいデータキャッシュ量を計算してその値を設定する指示を設定することにした。

#### 【0122】

I/Oベース性能748の性能値に関しては、I/O処理内容情報740cでは常にランダム性能（IOPS）である。この値の決定法は、例えば、実際に処理を実行し、そのときの処理スループットと記憶装置40のI/OパスI/F32のポート26のI/O性能を記憶装置40のモニタ情報から把握し、要求性能774に与えられるスループットを達成するのに必要なI/O性能をスループットとI/O性能が比例するとして見積り、その値を設定する。なお、複数のデータが同一のポート26経由で同一のLU208にデータを記憶している場合には、それらの間で必要な性能を適当に分配する。

#### 【0123】

次に、図16を用いてJobリソース管理情報620について説明する。

図16は、Jobリソース管理情報620のデータ構造を示す模式図である。

【0124】

Jobリソース管理情報620は、管理サーバ120のシステム管理プログラム140がワーク情報として利用する情報であり、Jobポート設定情報630とJobキャッシュ量情報650を含んでいる。

【0125】

Jobポート設定情報630は、記憶装置40のI/OパスI/F32のポート26毎の、ポート26を利用するJobとそのJobに割り当てられた優先アクセス制御機能の設定に関する情報であり、装置ID572を保持するエン트리と、ポートID384を保持するエン트리と、対応するポート26を利用してデータをアクセスするJobのJobID622とそのJobのそのポート26に対して割り当てられた性能割合632を保持するエントリの組を有する。なお、性能割合632は、ポート性能情報616に保持される対応ポートの最大性能からの割合として計算される性能指標値で、I/O処理量やデータ転送量への変換は、そのポート26に対応するポート性能情報618を参照して、そのポート26の最大I/O処理量や最大データ転送量を求め、それに性能割合632を乗じることによりおこなう。

【0126】

Jobキャッシュ量情報650は、記憶装置40のキャッシュグループ毎に、そのキャッシュグループに属するLU208を利用するJobとそのJobに対する割当量に関する情報であり、装置ID572を保持するエン트리、キャッシュグループID462を保持するエン트리、対応するキャッシュグループに属するLU208に記憶されているデータを利用するJobのJobID622を保持するエン트리、Jobへのキャッシュの割当量に関する情報である割当情報652を保持するエントリの組からなる。

【0127】

割当量に関しては、スタック形式で管理された複数世代の情報を持てるようにし、割当情報652中にその割当量が設定された理由に関しても割当量と組にし

て保持し、その原因が解決した場合には古い設定に戻れるようにする。なお、J o b が利用する一つ以上のキャッシュグループのキャッシュ割当量が変更される場合には、キャッシュ量の変更のない J o b が利用する他のキャッシュグループも同期して世代を更新する。

#### 【0128】

次に、図17を用いて優先度制御情報880について説明する。

図17は、優先度制御情報880のデータ構造を示す模式図である。

#### 【0129】

優先度制御情報880中は、管理サーバ120の中のシステム管理プログラム140がJ o b の優先度を考慮した制御をおこなう際に用いられる設定情報であり、J o b 優先度704を保持するエントリと、対応する余裕係数882を保持するエントリの組を含む。

#### 【0130】

本実施形態においては、J o b 優先度704は、P段階に分けられ、優先度の高いものから順にその値として1、2、…、Pを取る。余裕係数882とは、記憶装置40の設定変更方法を決定するとき、見積りデータに対して余裕を見込むための係数であり、J o b の優先度を高いものほど余裕を大きくするものであり、本実施形態では、1以上の値を取ることにしている。

#### (IV) 性能管理方法の処理

次に、図18ないし図27を用いて本実施形態の性能管理方法の処理について説明する。

#### 【0131】

##### (IV-1) システム管理プログラムの処理概要

先ず、システム管理プログラム140の処理概要について説明する。

#### 【0132】

システム管理プログラム140は、システムに関する構成を把握する。構成要素である記憶装置40、仮想化スイッチ60、サーバ70、ボリュームマネージャ78、ファイルシステム80、DBMS90、APプログラム100に関してそれぞれ識別子を付加して管理する。また、ネットワーク24やI/Oパス34

の接続関係をシステム管理情報 142 中に保持する。

【0133】

システム管理プログラム 140 は、他の装置から種々の情報を取得する。このとき、記憶装置 40 と仮想化スイッチ 60 が保持する情報に関しては、ネットワーク 24 を介して直接情報の送信要求を発行し、情報を取得する。また、サーバ 70 上で実行されるプログラムが保持する情報に関しては、ネットワーク 24 を介して管理エージェント 144 に対して情報の読み出し要求を発行して、それが必要な情報を読み出すことにより情報を収集する。

【0134】

システム管理プログラム 140 は、各仮想化機構から領域マッピング情報 300、DBMS 90 からデータ記憶領域情報 342、記憶装置 40 と仮想化スイッチ 60 から上位ポートマッピング情報 322、記憶装置 40 からキャッシュグループ情報 460 を取得し、それらを取得元の識別子とともにマッピング集約情報としてシステム管理情報 142 中に記憶する。

【0135】

ここで、システム管理プログラム 140 は、記憶装置 40 のキャッシュグループの管理において、ある特定のキャッシュグループをキャッシュ割当量の調整で優先的に削減対象とする「フリーキャッシュグループ」として扱う。キャッシュ量の割り当て要求がない LU 208 は、フリーキャッシュグループに属するとして制御する。

【0136】

また、ポートスループット設定情報 440 やキャッシュモニタ情報 362 に含まれる実領域分割数 502 と仮想管理領域数 504 に関しても各記憶装置 40 から取得し、それらの取得元の識別子とともにシステム管理情報 142 中に記憶する。これらの情報に関しては、設定変更後や一定期間経過後、情報の再取得と保持内容の整合性確認をおこなう。

【0137】

また、上記のシステム管理情報 142 のマッピング集約情報としてより詳細に説明すると、システム管理プログラム 140 は、実行中の Job のハードウェア

リソースに関連するマッピングの情報をシステム管理情報142中に保持する。サーバ70に関しては、Jobが実施されるサーバ70の識別子と、どのデータがどのポート26を経由してアクセスされるか、仮想化スイッチ60に関しては、データは、どの仮想化スイッチ60を経由し、仮想化スイッチ60のどのI/OパスI/F32のどのポート26を経由してアクセスされるか、記憶装置40に関しては、どのデータがどの記憶装置40のどのLU208を利用し、そのLU208は記憶装置40のI/OパスI/F32のどのポート26を経由してアクセスされ、そのLU208中のデータはどのHDD16に記憶されているか、また、そのLU208は、どのキャッシュグループに属するか、に関する情報を保持する。以下、この情報を「Jobリソースマッピング情報」と称する。

#### 【0138】

(IV-2) システム管理プログラムのJob管理方法とJob管理情報への値の設定

以下、システム管理プログラムのJob管理方法とJob管理情報への値の設定について説明する。

#### 【0139】

システム管理プログラム140は、バッチ業務を実施するためのJobの管理情報として、バッチJob管理情報700を、オンライン業務を実施するJobの管理情報として、オンラインJob管理情報770をシステム管理情報142中に、それぞれ保持する。これらの情報はHDD16に記憶され、システム管理プログラム140が起動時にメモリ14上のシステム管理情報142中に読み出される。

#### 【0140】

既に述べたように、Job管理情報700とオンラインJob管理情報770とを分けたのは、そのJobの性質により要求される性能、管理すべき情報が異なっていることを考慮したものである。

#### 【0141】

以下、バッチJob管理情報700とオンラインJob管理情報770のいずれか、または、その両方を指す場合に「Job管理情報」と称する。Job管理



情報は基本的に管理者が与えるが、一部の情報は後述する方法により設定される。

【0142】

また、バッチ Job 管理情報 700 中の I/O 処理内容情報 740 a とオンライン Job 管理情報 770 中の I/O 処理内容情報 740 c をまとめて、「I/O 処理内容情報 740」と称することにする。

【0143】

既に述べたように、本実施形態の Job の実行は、バッチ Job は連続して実行され、前段の Job の出力データを次段の Job の入力データとして引き継いでいくものとした。バッチ Job 管理情報 700 により管理される Job においては、その中の Job 依存出力データ情報 730 に設定されているデータに関し、Job の終了時に処理完了情報としてそれらの出力データ量を返すこととする。そして、この情報とバッチ Job 管理情報 700 中の情報を基に、連続的に実施される Job の実行時間見積りをおこなう。

【0144】

また、本実施形態において、出力データ量は、バッチ業務向け Job が実行する AP プログラム 100 が処理完了情報として出力するものを利用する。

【0145】

以下、バッチ Job 管理情報 700 の値の設定処理について説明する。

【0146】

バッチ Job 管理情報 700 は、バッチジョブの Job 毎に与えられる情報であった。

【0147】

このエントリの中で、Job ID 622、Job 優先度 704、Job 種別 706、実行条件 708、最大実行時間 710、実行サーバ ID 712、実行コマンド 714、Job 依存入力情報 720、Job 依存出力データ情報 730、予め管理者から Job 実行前に与えられる情報である。

【0148】

なお、Job の最大実行時間を見積る方法については、後に詳細に説明する。

## 【0149】

CPU処理時間見積方法716とI/O処理内容情報740aに関しては、モニタ情報を実測して設定される。

## 【0150】

以下、図18を用いて、このCPU処理時間見積方法716とI/O処理内容情報740aを設定する処理について説明する。

図18は、Job管理情報700中のCPU処理時間見積方法716とI/O処理内容情報740aをモニタ情報の実測により設定する処理のフローチャートである。

## 【0151】

手順開始時、バッチJob管理情報700中の他の情報と、図14に示したバッチSQL設計情報850が管理者から与えられる。なお、この手順を開始する際には実際の動作時と同じように各種データがシステム内に存在する必要がある（ステップ1101）。

## 【0152】

まず、バッチSQL設計情報850中の実行DBMSID852から処理を実施するDBMS90を把握し、そこから実行SQL文862中のSQL文のSQL実行計画を全て取得する（ステップ1102）。

## 【0153】

SQL実行計画は、SQLを実行するための情報であり、SQL文を実施するための処理ステップとその処理内容、利用データとアクセス方法の詳細情報などを含んでいる。

## 【0154】

なお、処理中に明示的に一時的に利用する表を作成する必要があるが、その表を利用するSQL文のSQL実行計画取得には、その表を作成した後にSQL実行計画を取得する。可能なら、処理をSQL文毎にステップ実行し、その処理実行直前にSQL実行計画を取得する。全てのSQL実行計画を取得した後に一時的に利用する表を削除する。

## 【0155】

取得したSQL実行計画とバッチSQL設計情報850中の情報から、各SQL文によりアクセスされるデータとアクセス方法、アクセス順を把握する（ステップ1103）。

【0156】

SQL実行計画は、SQL文を実施するための処理ステップとその処理内容、利用データとアクセス方法の詳細情報を含むため、一つのSQL文内でのアクセスデータ・アクセス方法・アクセス順は、SQL実行計画から把握できる。

【0157】

処理実施の際にDBMS90により暗黙に作成される一時表に関しては、その表が作成されるデータ領域のデータ領域名344を把握する。バッチSQL設計情報850中のSQLID858と繰り返し実行SQL情報862から、繰り返しを含めた実行順序がわかる。このとき、一つ以上のSQL文が繰り返し実行される場合には、それらがアクセスするデータは繰り返し処理により全て同時にアクセスされるものとする。

【0158】

そして、把握したアクセス先データとアクセス順をI/O処理内容情報740a中のデータ情報724とアクセス順742、I/Oベース性能748のポートID384に設定する。同じデータに複数の異なる処理ステップでアクセスされる場合には、それぞれ独立したものとする。ポートID384は、システム管理情報140中のマッピング集約情報を参照することにより、それが存在する記憶装置40と共に求めることができ、複数のポートID384が求まる可能性もある。

【0159】

ここで、把握した処理が“表の全走査”の場合にはI/Oタイプ746は“シーケンシャル”（図13では、「Seq」）に、B-T r e e索引を利用した選択処理”の場合には、索引と表の双方ともに“ランダム”（図13では、「Rnd」）に設定する。ファイル202に対する入出力は全て“シーケンシャル”とする。

【0160】

これから処理を実施する際に外部から入力されるデータ量を把握する（ステップ1104）。

【0161】

J o b 依存入力情報720中に設定されているデータに関して、そのデータ量単位726に従ってデータ量を把握する。本実施形態では、原則、複数のJ o bが一連の処理を形成するため、それらを順次実行して完了報告としてシステム管理情報142中に出力された出力データ量を把握する。

【0162】

続いて、実際に処理を実行し、実行時間の計測と、DBMS90が稼動するサーバ70におけるCPU稼働率と、記憶装置40におけるI/O処理実行状況（I/O処理実行回数、データ転送量）を把握する（ステップ1105）。

【0163】

それらの稼動状況は、前述のシステム管理プログラム140が収集することにより把握する。このデータ取得の際にデータ取得間隔は処理の実行時間に比して十分に小さなものとする。システム負荷を削減するため、可能であれば、処理実行に関連するサーバ70のCPU稼働率とステップ1103で把握したデータを記憶している部分のみ詳細なモニタ情報を取得する。

【0164】

取得したモニタ情報から、DBMS90が稼動するサーバ70におけるCPU稼働率がある閾値（例えば、80%）以上の時間を把握し、その時間はバッチSQL設計情報850中の主入力データ854で指示されるデータのデータ量に比例するものとしてCPU処理時間見積方法716を決定する（ステップ1106）。

【0165】

入力データ量は、ステップ1104で把握したものをを用いる。

【0166】

次に、モニタ情報から各アクセスデータにおけるI/Oネック、I/Oベース性能、入力データ量あたりのI/O処理量を把握し、I/O処理内容情報の値設定をおこなう（ステップ1107）。

## 【0167】

すなわち、ステップ1103で把握したアクセスデータとアクセス順、記憶装置40のアクセス先のポート26を基に、記憶装置40のストレージポートモニタ情報382中のポート26のモニタ情報との対応付けをおこなう。

## 【0168】

その後、あるアクセスデータに対応する期間における、DBMS90が稼働するサーバ70のCPU稼働率のモニタ値を調べ、それがステップ1106の閾値を越えている場合にはI/Oボトルネックではないとして対応するI/Oネック744の値を“Y”に設定し、そうでない場合には“N”に設定する。

## 【0169】

続いて、モニタ情報からポート26におけるI/O処理量とI/O処理を実施していた時間をLU208毎に把握し、その平均性能としてデータに対応するI/Oベース性能748をポート26毎に設定する。このとき、対応するI/Oタイプ744の値が“シーケンシャル”の場合には処理量としてデータ転送量を、“ランダム”の場合にはI/O処理実行回数を用いる。さらに、ステップ1104で把握した入力データ量とポート26毎のI/O処理量から、対応する処理量見積752を設定する。なお、同一のLU208に複数の異なるデータが記憶され、それらが同一のポート26を経由してアクセスされる場合には、それらの間で適当にI/O処理量を分割する。

## 【0170】

そして、手順を終了する（ステップ1108）。

## 【0171】

本手順では、I/O処理内容情報740a中のストレージヒント754は設定されない。ここは、本手順終了後に管理者が設定する。ストレージヒント754を設定した場合には、I/O処理性能が変化するため、ステップ1105からステップ1107を再度実行し、I/Oネック744とI/Oベース性能748を再設定する。

## 【0172】

次に、バッチJob管理情報700で管理されるJobの最大実行時間710

の処理時間の見積り方法について説明する。

【0173】

本実施形態においては、バッチ J o b 管理情報 7 0 0 で管理される J o b の処理時間を、C P U がボトルネックとなっている状態での C P U 処理時間と、I / O がボトルネックとなっている状態での I / O 処理時間と、C P U がボトルネックとなっている状態での I / O 遅延による影響時間の和として見積る。また、各 J o b への入力データ量は、システム管理情報 1 4 2 中に記憶されるものとする。

【0174】

C P U がボトルネック状態での C P U 処理時間は、C P U 処理時間見積り方法 7 1 6 と処理開始時に与えられる入力データ量から計算する。

【0175】

I / O がボトルネック状態での I / O 処理時間は、I / O 処理内容情報 7 4 0 a 内の I / O ネットワーク 4 4 4 の値が“Y”であるデータ情報 7 2 4 で識別されるデータに関して、処理開始時に与えられる入力データ量と処理量見積 7 5 2 により求まる処理量と、データアクセスに利用可能な I / O 処理性能から計算し、その総和として求める。

【0176】

各データへのアクセスに利用可能な処理性能は以下のように求める。まず、J o b がアクセスする各データに関して、それをアクセスする際に利用される記憶装置 4 0 の I / O パス I / F 3 2 のポート 2 6 を I / O ベース性能 7 4 8 から求める。続いて、各データ毎に求めたポート 2 6 とそのデータに対応するアクセス順 7 4 2 の値によりデータをグループ化し、ポート 2 6 を同時にアクセスするデータの組を求める。それぞれのグループにおいて、ポート 2 6 で J o b が利用可能な性能割合 6 3 2 は、J o b ポート設定情報 6 3 0 に保持されているか、あるいは、別途条件として指定されている。そして、その性能割合 6 3 2 を各データに対応する I / O ベース性能 7 4 8 の対応ポート 2 6 における性能値から計算される性能割合 6 3 2 に比例して分配する。このようにして、あるデータが利用可能な性能割合 6 3 2 を求め、これからデータへのアクセスに利用可能なポート 2

6におけるI/O処理性能を求める。

【0177】

CPUがボトルネックとなっている状態でのI/O遅延による影響時間は次のように求める。I/O処理内容情報740a内のI/Oネック444の値が“N”であるデータ情報724で識別されるデータに関して、まず、I/Oがボトルネック状態でのI/O処理時間と同様に現在のI/O処理量設定での処理時間を求める。引き続き、I/O処理性能がI/Oベース性能748に与えられた性能であると仮定して同様にI/O処理時間を求める。各データのポート26毎に前者から後者を引いた値が求める影響時間である。なお、計算により負の値が求めた場合には影響時間は0であるとする。

【0178】

本実施形態においては、入力データ量は、原則前段のJobの処理完了報告から把握する。常時実行中のAPプログラム100がある期間内に処理したデータを抽出するようなJob等、対応する前段Jobが存在しない場合、Jobへの入力データ量を取得するためのJobを作成・先行して実施させ、それに目的とする入力データ量を完了報告として出力させることにより、処理時間の見積りに必要となる入力データ量がシステム管理情報142中に記憶されるようにする。前述の例では、入力データ量として、APプログラム100の対象処理の対象期間内の処理実行回数をモニタ履歴情報510から取得し、それを件数として利用する。なお、データ量見積りJobを作成する代わりに、Job依存入力情報720中にデータ入力量を取得する方法を記憶させ、それを用いてデータ量を把握してもよい。

【0179】

(IV-3) システム管理プログラムのモニタ情報の収集とそのモニタ情報によるJobチューニング処理

次に、図19ないし図21を用いてシステム管理プログラムのモニタ情報の収集とそのモニタ情報によるJobチューニング処理について説明する。

【0180】

システム管理プログラムのモニタ情報の収集とそのモニタ情報によるJobチ

ューニング処理は、管理サーバ120の実行中でシステム管理プログラムがシステムを監視している間は、常時、無限ループで実行されることになる。

図19は、システム管理プログラムのモニタ情報の収集とそのモニタ情報によるJobチューニング処理のゼネラルチャートである。

図20は、モニタ情報によるJobチューニング処理のフローチャートである（その一）。

図21は、モニタ情報によるJobチューニング処理のフローチャートである（その二）。

#### 【0181】

モニタ履歴情報510への値の設定は以下のようにおこなう。

#### 【0182】

この処理は、管理サーバ120が起ち上がり、システム管理プログラム140が起動し、その初期化が完了したときに開始される（ステップ1401）。

#### 【0183】

先ず、初期モニタ値として、モニタ情報を取得している全ての装置やプログラムから現在のモニタ情報の値を取得し（ステップ1402）、前回取得情報518中の対応する取得日時と取得値のエントリを更新する。

#### 【0184】

続いて、次のデータ取得タイミングまで、処理をスリープさせる（ステップ1403）。なお、データの取得は、前回モニタ値取得後一定間隔、もしくは、事前に定めた特定時刻におこなう。このとき、モニタする内容によりデータの取得間隔を個別に管理し、データ取得間隔をモニタする内容毎に異なるものとしてもよい。

#### 【0185】

データを取得するタイミングが来たときに、現在のモニタ情報の値と現在日時を取得する。そして、それら情報の取得後、対応する前回取得情報518中に保存された前回取得日時とその取得値を把握し、履歴内容522の形の値を計算する方法をモニタ内容516を基に選択し、その値を計算する（ステップ1404）。



## 【0186】

現在日時と前回取得日時から期間を把握し、計算値を期間の平均かつ最大としてサンプル524を保持するエントリに記憶する。その後、現在日時と今回取得したモニタ情報の値を前回取得日時とその取得値として前回取得情報518に記憶する。

## 【0187】

続いて、ステップ1501から始まる収集したモニタ情報を基にしたチューニング処理を実施する（ステップ1405、図20、図21の処理）。

## 【0188】

そして、ある一定時間以前のデータを保持するサンプル524を整理して、モニタ履歴情報510のデータ量を削減する（ステップ1406）。

## 【0189】

その方法は、データの古さに応じて、1）ある期間内の複数のサンプル524を一つのサンプル524にまとめる、2）特に古いサンプル524のデータを削除する、ことをおこなう。処理完了後ステップ1403へ戻り、以下、繰り返して処理を実施する。

## 【0190】

各装置の構成が変更されることにより、モニタすべき部位が動的に増減する。モニタ部位が増えた場合には、ステップ1404でモニタ履歴情報510中にエントリを追加し、その部位もモニタする。モニタ部位が削除された場合には、それに対応するデータは、ステップ1404の処理をおこなわず、ステップ1406において削除されたモニタ部位のサンプル524が全て削除された場合に、モニタ履歴情報510中のエントリを全て削除する。

## 【0191】

次に、図20および図21を用いてモニタ情報によるJobチューニング処理について説明する。

## 【0192】

このモニタ情報によるJobチューニング処理は、図19のステップ1405に示された処理である。

## 【0193】

この処理は、実行中の全 J o b について、各ステップを同時に、ステップ内では J o b 優先度 704 が最も高いものから順に処理を実施する。ステップ 1515、ステップ 1517において、複数の J o b により同一の部位を対象としてデータの移動や I/O 処理量の増加/削減が要求される場合には、J o b 優先度 704 が最も高い J o b についてのみ処理を実施する。また、一つの J o b に複数の問題点が存在すると判断された場合には、それぞれ独立した複数のパスに分岐する。

## 【0194】

J o b チューニング処理は、図 19 のステップ 1405 で、サブルーチンとしてコールされる (ステップ 1501)。

## 【0195】

先ず、システム管理 142 中に記憶されている「データ移動要求」により HDD 過負荷状態の変更が実行可能か否かを確認する (ステップ 1502)。HDD 過負荷状態の変更とは、データ移動によって、LU208 を適当な HDD16 に配置換えするものである。なお、ここで扱うデータ移動要求は、以前に実施された J o b チューニング処理のステップ 1508 (後述) で実施されたものである。

## 【0196】

ここで、直近と、最近のある一定期間 (例えば 1 週間) 内の現時刻から一定時間 (例えば 30 分間) における移動元 HDD16、移動先 HDD16 の全体の稼働率の最大値をモニタ履歴情報 510 を用いて把握し、双方でその値が事前に定められた閾値 (例えば 40%) を下回っている場合にデータ移動を実行可能と判断する。他の仮想化機構におけるデータ移動の要求が存在する場合にも、同様に関連する部位の稼働率を調べ、全ての部位でその値が事前に定められた閾値未満のときに、データ移動とそれに伴う設定の変更を実施する。実行可能と判断されたデータ移動要求に関し、関連する仮想化機構に対して、そのデータ移動を実現するための処理要求を発行する。

## 【0197】

次に、J o b が現在、記憶装置 40 のデータキャッシュ 42 に関して、初期設定よりも多くのキャッシュ量を利用している場合に、それを開放できるか確認する（ステップ 1503）。

## 【0198】

ここでは、図 16 に示した J o b キャッシュ量情報 650 中の割当情報 652 を参照し、設定量が初期設定（最も古い世代のデータ）より大きなエントリを新しい世代の方から検索し、その原因をその設定理由から把握する。そして、その原因に対応するモニタ履歴情報 510 の最新のデータを把握し、設定理由中に記憶された負荷のよりも一定割合（例えば 75%）未満の負荷になっているか確認し、負荷が下がっている場合には、その 1 世代前の設定に戻すと判断する。前の設定に戻すと判断された場合には、割当情報 652 のさらに古い世代のデータの確認を同様に繰り返しおこなう。

## 【0199】

この確認処理により、戻されるべき割当情報 652 の世代を把握し、そのようなものが存在する場合には、J o b キャッシュ量情報 650 中の割当情報 652 の戻される設定以降の新世代のデータを破棄し、戻された世代のデータに合わせて、記憶装置 40 に対してキャッシュ割当量の変更指示を発行する。

## 【0200】

次に、モニタ履歴情報 510 の最新データを参照し、問題点が存在するか確認する（ステップ 1504）。

## 【0201】

ここで、J o b がオンライン J o b 管理情報 770 で管理されている場合、すなわち、オンライン J o b である場合には、図 12 に示されたモニタ履歴情報 510 に記憶された、最新の A P プログラム 100 における各処理の平均応答時間、平均処理時間、平均実行回数を把握し、それから A P プログラム 100 全体での平均応答時間、平均稼働率、平均実行回数を計算する。そして、対応するオンライン J o b 管理情報 770 を参照し、求めた平均応答時間が要求性能 774 に設定された平均応答時間を満たすか、また、平均稼働率が（処理多重度 782）×（事前に定められた閾値）（例えば 95%）を超えている場合に高負荷状態に

あるとし、平均実行回数が要求性能774に設定された処理スループットを満たすか否かを確認して、満たさない条件が存在する場合に問題点が存在するとする。

#### 【0202】

J o bがバッチJ o b管理情報700で管理されている場合、すなわち、バッチJ o bである場合には、J o bリソースマッピング情報を参照してそのJ o bが利用するデータが記憶されるHDD16を把握し、それらのHDD16全体の稼働率が事前に定められる閾値（例えば70%）を超えている場合には過負荷状態にあると判断し、問題点が存在するとする。

#### 【0203】

問題点が存在する場合には、ステップ1505に進む。存在しない場合には、ステップ1530に進み、処理を終了する。

#### 【0204】

次に、図12に示されたモニタ履歴情報510の最新データを参照し、J o bに関連するハードウェアに関して、記憶装置40以外に過負荷状態の部位が存在するか確認する（ステップ1505）。

#### 【0205】

J o bに関連するハードウェアとしては、サーバ70におけるCPU稼働率、仮想化スイッチ60とサーバ70におけるI/OパスI/F32のポート26等があり、それらはJ o bリソースマッピング情報を参照することにより求めることができる。過負荷状態とは、各々の稼働率が各ハードウェア毎に事前に定められた値を超えた状態として認識する。そのような部位が存在する場合には、ステップ1525に進み、存在しない場合にはステップ1506に進む。

#### 【0206】

ステップ1505で過負荷状態であると判断された部位の負荷を下げる構成を取ることが可能であれば、それを実現するようにサーバ70中のOS72や仮想化スイッチ60に対して指示を出す（ステップ1525）。なお、本発明は、記憶装置の性能を調整することに主眼を置いているので、ステップ1525の具体的な方法については詳説しない。

## 【 0 2 0 7 】

ステップ 1 5 0 5 で記憶装置 4 0 以外に過負荷状態の部位が存在しないと判断されたときには、J o b が利用するデータを記憶する記憶装置 4 0 内の関連部位に関して、過負荷状態のものが存在するか否かを確認する（ステップ 1 5 0 6）。

## 【 0 2 0 8 】

確認部位としては、I / O パス I / F 3 2 のポート 2 6 と HDD 1 6 がある。HDD 1 6 についてはステップ 1 5 0 4 で述べた通りである。

## 【 0 2 0 9 】

ポート 2 6 に関しては、J o b がオンライン J o b 管理情報 7 7 0 で管理される場合のみ確認する。まず、システム管理情報 1 4 2 の中に保持する J o b リソースマッピング情報を参照し、J o b が利用するポート 2 6 とそこを通してアクセスする LU 2 0 8 を把握する。図 1 2 に示したモニタ履歴情報 5 1 0 の最新データを用いて、求めた LU 2 0 8 のポート 2 6 における性能割合 6 3 2 を計算し、ポート 2 6 毎にその総和を求める。求めた値が、J o b ポート設定情報 6 3 0 に保存されている J o b に対して割り当てられているポート 2 6 の性能割合 6 3 2 の一定割合（例えば、9 0 %）を超える場合に、そのポート 2 6 は J o b に対し過負荷状態にあると判断する。

## 【 0 2 1 0 】

ステップ 1 5 0 6 で、関連するポート 2 6 が J o b に対し過負荷状態にある場合にはステップ 1 5 1 1 に進む（図 2 1 [A]）。関連する HDD 1 6 が過負荷状態にある場合にはステップ 1 5 0 7 へ進む。そして、関連する全てのポート 2 6 と HDD 1 6 が正常値と判断される場合には、ステップ 1 5 1 9 に進む。なお、このステップ以降は、過負荷状態の部位が複数存在する場合には、その部位毎にパスが分かれて、それぞれについて処理が進められる。

## 【 0 2 1 1 】

ステップ 1 5 0 6 で、関連する全てのポート 2 6 と HDD 1 6 が正常値と判断される場合には、性能問題が発生している J o b が利用するキャッシュグループのキャッシュ量割当を変更する方法を決定する（ステップ 1 5 1 9）。

## 【 0 2 1 2 】

この処理では、このステップに来た J o b に関して、J o b 優先度 7 0 4 が高い順に、オンライン J o b 向けにキャッシュ量をチューニングする処理を実施する。なお、オンライン J o b 向けにキャッシュ量をチューニングする処理は、後に (IV-5) の図 2 5 で説明する。

## 【 0 2 1 3 】

このとき、先にチューニングを実施した J o b が利用するキャッシュグループをシステム管理情報 1 4 2 の中の J o b リソースマッピング情報から把握し、後続の J o b に関して、オンライン J o b 向けにキャッシュ量をチューニングする処理を実施する際に、それらキャッシュグループのキャッシュ割当量を減らさないように指示を出す。なお、J o b に対する現在のキャッシュ割当量の一定比率分（例えば 5 %）をフリーキャッシュグループから割り当てた後に、オンライン J o b 向けにキャッシュ量をチューニングする処理を実施してもよい。

## 【 0 2 1 4 】

そして、ステップ 1 5 1 9 で求めたキャッシュ割当量の変更指示を対応する記憶装置 4 0 に発行し、その設定に合わせて、図 1 6 の J o b リソース管理情報 6 2 0 の中の J o b キャッシュ量情報 6 5 0 の値を更新する（ステップ 1 5 2 1）。

## 【 0 2 1 5 】

そして、処理を終了する（ステップ 1 5 3 0）。

## 【 0 2 1 6 】

ステップ 1 5 0 7 で関連する HDD 1 6 が過負荷状態にあると判断された場合には、J o b が過負荷状態にある HDD 1 6 にデータを記憶する LU 2 0 8 に属するキャッシュグループにキャッシュを追加するチューニングが実行可能か確認する（ステップ 1 5 0 7）。

## 【 0 2 1 7 】

そのようなキャッシュグループに対して、その J o b のキャッシュ量を現在値からある一定割合（例えば 5 %）増加させるとしたとき、フリーキャッシュグループからその量を確保できるか J o b キャッシュ量情報 6 5 0 を参照して確認し

、割当可能な場合にはそのキャッシュ割当量変更処理を実施することとする。不可能な場合には、可能量を全てこのチューニングに用いる。このとき、J o b が利用する異なるデータについて、それぞれの異なる L U 2 0 8 により問題の H D D 1 6 に記憶される場合には、用いた L U 処理時間変化の平均を考え、関連する L U 2 0 8 におけるその和が最大になるように割り当てる。なお、L U 処理時間変化の平均を考えて、キャッシュ割当量変更変更案を作成する処理については、後の (IV-5) の図 2 5 で説明する。また、J o b キャッシュ量情報 6 5 0 の割当情報 6 5 2 をこのステップで設定した値に更新するときには新しい世代のデータとして記憶し、設定理由に過負荷 H D D 1 6 の H D D I D 3 9 4 と現在の稼働率を記憶する。

## 【0218】

次に、過負荷状態の H D D 1 6 上に記憶されている、J o b が利用する L U 2 0 8 のデータを通常負荷の H D D 1 6 へ移動させるデータ移動要求を作成する (ステップ 1 5 0 8)。

## 【0219】

データ移動要求は、以下のようにして作成される。まず、過負荷状態の H D D 1 6 を移動元 H D D 1 6 とする。J o b リソースマッピング情報を参照し、移動元 H D D 1 6 に記憶された L U 2 0 8 のうち、そのデータを利用する最も優先度の高い J o b が利用する L U 2 0 8 をデータを移動する L U 2 0 8 として把握する。そして、移動元 H D D 1 6 を有する記憶装置 4 0、移動元 H D D 1 6 データを移動する L U 2 0 8 を指定して、H D D 過負下状態を解消する方法を決定する処理を実施してデータ移動要求を作成し、システム管理情報 1 4 2 中に記憶する。H D D 過負下状態を解消する方法については、後に (IV-5) の図 2 7 で説明する。

## 【0220】

データ移動要求作成後に、それを実行した後の稼働率を、移動した L U 2 0 8 の稼働率分だけ H D D 1 6 全体の稼働率が下がるとして見積り、その値が事前に定めた閾値 (例えば 6 0 %) を下回らない場合には、次に優先度の高い J o b が利用する L U 2 0 8 をデータを移動する L U 2 0 8 として、H D D 過負下状態

を解消する方法を決定する処理を実施し、移動元HDD16の稼働率見積値が閾値を下回るか、あるいは、全てのJ o bに関してその確認を完了するまで繰り返す。

#### 【0221】

なお、仮想化スイッチ60やボリュームマネージャ78がデータ移動により仮想構造と管理構造との対応関係を動的に変更できる場合には、記憶装置40内のデータ移動要求の作成に失敗した場合に、HDD過負下状態を解消する方法を決定する処理からの処理を準用して仮想化スイッチ60やボリュームマネージャ78におけるデータ移動の要求を作成してもよい。このとき、以下の点も考慮する。1) 新たに仮想構造をフリー領域から作成してそこにデータを移動させる。2) 移動先決定時にHDD16の稼働率のみでなく、データ移動先に対するアクセスパス(サーバ70から記憶装置40へ至るI/Oパス34の経路)におけるポート26の稼働状況を調べ、それぞれで移動後の値として見積られる稼働率が100%を超えないようにする。この見積りは、HDD16における稼働率の見積りと同様に実施する。3) 移動先となる記憶装置40において、移動されたデータを保持するLU208が属するキャッシュグループに対して、現状と同じ分のキャッシュ量を割り当てる。

#### 【0222】

次に、ステップ1507で割り当てたキャッシュ量が十分か判断する(ステップ1509)。

#### 【0223】

HDD過負下状態を解消する方法で決定されたLU処理時間変化の平均を考え、関連するLU208におけるその値の和が事前に定められた閾値(例えば単位時間の5%)より大きい場合に十分と判断し、ステップ1521に進む。そうでない場合は不十分としてステップ1517に進む。

#### 【0224】

ステップ1507で割り当てたキャッシュ量が十分と判断されたときには、これまでに求めたキャッシュ割当量の変更指示やI/O性能の設定変更を対応する記憶装置40に発行し、その設定に合わせて、図16のJ o bリソース管理情報



620の中のJobポート設定情報630とJobキャッシュ量情報650の値を更新する(ステップ1521)。

#### 【0225】

ステップ1507で割り当てたキャッシュ量が不十分と判断されたときには、ここでは、過負荷状態にあるHDD16にJobがデータを記憶するLU208へのI/O処理量を削減するチューニングが可能か判断し、可能ならI/O処理の設定変更をおこなう(ステップ1517)。なお、本ステップは、ポートにも問題があるときに、別パスでこの処理とは非同期に実施されるステップ1515より後に実施される必要がある。

#### 【0226】

この処理では、まず、現在チューニング中のJobのJob管理情報からJob優先度704を求める。過負荷状態にあるHDD16にデータを記憶するLU208を利用するJobを、システム管理情報124の中のJobリソースマッピング情報を参照して把握し、そのJobのJob優先度704を前述と同様に求め、先に求めた現在チューニング中のJobのJob優先度704より低いものを把握する。それらのJobに関して、過負荷状態にあるHDD16にデータを記憶するLU208をアクセスする際に利用される記憶装置40のI/OパスI/F32のポートをJobリソースマッピング情報から求め、Jobポート設定情報630に記憶された、そのJobのそのポート26における性能割合632の値を事前に定められた割合(例えば現在の値5%)を削減することとする。なお、直接HDD16へのアクセス量を制限できるときには、その設定をおこなってもよい。

#### 【0227】

そして、ステップ1521を実施し、処理を終了する(ステップ1530)。

#### 【0228】

ステップ1506で、関連するポート26がJobに対し過負荷状態にある場合には、ポート26がJobに対し過負荷状態である場合の解消方法を決定する(ステップ1511)。まず、Jobに対して過負荷状態にあるポート26において、図16に示したJobリソース管理情報620の中のJobポート設定情

報 630 を参照して、J o b に対して割り当てられているポート 26 の性能割合 632 を求め、その値を一定割合（例えば 5 %）増やす I / O 性能設定変更方法を作成する。

#### 【0229】

次に、ステップ 1511 で作成した I / O 処理量増加案を実現可能か確認する（ステップ 1512）。

#### 【0230】

ステップ 1511 で求めた I / O 性能設定変更方法を採用した際に、J o b に対して過負荷状態にあるポート 26 の各 J o b に対して割り当てられた性能割合 632 を、J o b ポート設定情報 630 を参照して求めてその和を計算し、その値が 100 % を超えない場合には実現可能としてステップ 1521 に進む。100 % を超える場合には実現不可能としてステップ 1513 に進む。

#### 【0231】

ステップ 1512 で、I / O 処理量増加案が実現不可能と判断されたときには、J o b に対し過負荷状態にある問題ポート 26 経由してアクセスされる LU208 に対して、サーバ 70 からそれを回避してアクセスするアクセスパスを追加するか、変更することにより、過負荷状態を回避することが可能か否かを確認する（ステップ 1513）。

#### 【0232】

これは、ボリュームマネージャ 78、仮想化スイッチ 60、記憶装置 40 が動的に複数のアクセスパスを設定・変更することにより実現するもので、J o b リソースマッピング情報やその他システム管理情報 142 中に保持される情報を用いて確認する。可能であるときにはステップ 1523 に進み、不可能である場合にはステップ 1514 に進む。

#### 【0233】

ステップ 1513 で、アクセスパスを変更・追加可能と判断されたときには、ステップ 1513 で作成または変更可能と判断された、問題となっているポート 26 経由してアクセスされる LU208 に対して、サーバ 70 からそれを回避してアクセスするアクセスパスを実現するように、ボリュームマネージャ 78、仮

想化スイッチ 6 0、記憶装置 4 0 に対して指示を出して（ステップ 1 5 2 3）、処理を終了する（図 2 0 [C]、ステップ 1 5 3 0）。

【 0 2 3 4 】

ステップ 1 5 1 3 で、アクセスパスを変更・追加不可能と判断されたときには、次に、J o b に対し過負荷状態のポート 2 6 は複数の J o b により共有されているか J o b リソースマッピング情報を参照して確認する（ステップ 1 5 1 4）。

【 0 2 3 5 】

共有されている場合にはステップ 1 5 1 5 に進む。共有されていない場合には、チューニング不可能としてステップ 1 5 3 0 に進み、処理を終了する。

【 0 2 3 6 】

ステップ 1 5 1 4 で、過負荷状態のポート 2 6 は複数の J o b により共有されていると判断されたときには、問題となっている J o b のうち最も J o b 優先度 7 0 4 が高いものに対し、過負荷状態のポート 2 6 を共有する他 J o b の I / O 処理量を絞り込む I / O 処理性能の設定変更方法を決定する（ステップ 1 5 1 5）。

【 0 2 3 7 】

すなわち、問題ポート 2 6 で現在 J o b に割り当てられている性能割合 6 3 2 の値を一定割合（例えば 5 %）増加させるとし、J o b の J o b I D 6 2 2、問題ポート 2 6 のポート I D 3 8 4、新たに割り当てる性能割合 6 3 2 を指定して、ステップ 1 6 0 1 から始まる処理を実施し、そのポート 2 6 における新たな I / O 処理性能の割当を決定する。

【 0 2 3 8 】

そして、ステップ 1 5 2 1（図 2 0 [B]）に進む。

【 0 2 3 9 】

（IV-4）システム管理プログラムによる J o b 実行処理

次に、図 2 2 を用いてシステム管理プログラムによる J o b 実行処理について説明する。

図 2 2 は、システム管理プログラムによる J o b 実行処理のフローチャートで

ある。

【0240】

システム管理プログラム140は、与えられたJob管理情報中の実行条件708を参照して、与えられた条件が満たされた場合にそのJobを実行する（ステップ1301）。

【0241】

そのときに、処理開始時の時刻をJobID622とともにシステム管理情報142中に記憶する。なお、複数のバッチ業務向けのJobが連続して実行される場合には、前段の処理時間によりバッチJob管理情報700中の実行最大時間710の値を変更してもよく、その場合には元の値を別途保存する。

【0242】

Jobが開始されると、まず、Jobが利用するハードウェアリソースを把握し、Jobリソースマッピング情報に設定する（ステップ1302）。

【0243】

これは、Job管理情報中の実行サーバID712から処理が実施されるサーバ70、I/O処理内容情報740中のデータ情報724から利用するデータを把握し、それらによりシステム管理情報142中のマッピング集約情報を参照しておこなう。

【0244】

次に、優先アクセス制御機能を利用した記憶装置40のI/OパスI/F32のポート26におけるI/O処理性能の変更方法を決定して、それに基づいて記憶装置40に対してその設定指示を出す（ステップ1303）。I/O処理性能の変更方法の決定処理については、後に（IV-4）で説明する。

【0245】

ここでは、開始するJobへ割当を追加する条件で、I/O処理性能の変更方法の決定処理を実施し、求めた値をJobポート設定情報630に反映させる。ここで求まる変更案は、Job毎の割当であり、システム管理情報142の中のJobリソースマッピング情報を利用して、記憶装置40へのアクセス元となるサーバ70もしくは仮想化スイッチ60のI/OパスI/F32のポート26を

求め、同一のポート 2 6 からのものをまとめることにより記憶装置 4 0 への設定値を求め、記憶装置 4 0 に対してその設定指示を出す。

## 【 0 2 4 6 】

次に、キャッシュグループの設定とそのキャッシュ量の設定・変更方法を決定すし、それに基づいて記憶装置 4 0 に対してその設定指示を出す（ステップ 1 3 0 4）。J o b を開始する際のキャッシュ量の割当てを決定する処理は、後に（IV-5）の図 2 3 で説明する。

## 【 0 2 4 7 】

ここでは、先ず、J o b 管理情報の I / O 処理処理内容情報 7 4 0 中のストレージヒント 7 5 4 からキャッシュ量の割当指示があるか調べる。続いて、割当指示があるデータに関して、J o b リソースマッピング情報を参照し、それが記憶される L U 2 0 8 を調べ、それがフリーキャッシュグループに属しているかを調べる。属すものが存在する場合には、同じデータを記憶し、かつ同じ記憶装置 4 0 により提供される L U 2 0 8 は同じキャッシュグループとなるような構成案を作成し、必要であればキャッシュグループの新規作成をおこなうことにし、J o b リソースマッピング情報にその情報を追加する。このとき、できる限り異なるデータを保持する L U 2 0 8 は異なるキャッシュグループに属するようにする。

## 【 0 2 4 8 】

続いて、開始する J o b の J o b 管理情報とキャッシュグループ設定を変更する場合にはその情報を与えてステップ 2 3 0 1 から始まる処理を実施し、J o b キャッシュ量情報 6 5 0 に値を設定する。求まる値は J o b 毎のキャッシュ割当量であり、あるキャッシュグループが利用するキャッシュ容量はそのキャッシュグループを利用する J o b に与えられた割当量の総和とする。それらの変更内容に従って記憶装置 4 0 に対して指示を出す。

## 【 0 2 4 9 】

これらの処理の後、J o b 管理情報中の実行サーバ I D 7 1 2 で識別されるサーバ 7 0 に、対応する実行コマンド 7 1 4 を実行するように指示を出し、処理を実行する（ステップ 1 3 0 5）。

## 【 0 2 5 0 】

処理完了後、バッチ J o b 管理情報 7 0 0 により管理される J o b の場合には、J o b 依存出力データ情報 7 3 0 に設定されているデータの出力データ量を処理完了情報として取得し、その値を J o b I D 6 2 2 やその出力データを識別するデータ情報 7 2 4 とともにシステム管理情報 1 4 2 内に記憶する。

## 【 0 2 5 1 】

J o b が終わると、ステップ 1 3 0 3 で設定したポート 2 6 における I / O 処理性能の割当を他の J o b 向けに開放する（ステップ 1 3 0 6）。

## 【 0 2 5 2 】

ここでは、実行完了した J o b への割当分を削除する条件で、ステップ 1 3 0 3 と同様の処理を実施する。

## 【 0 2 5 3 】

次に、ステップ 1 3 0 4 で設定したキャッシュグループとキャッシュ割当の設定を解除する（ステップ 1 3 0 7）。

## 【 0 2 5 4 】

ここでは、完了した J o b の J o b I D 6 2 2 を与えてステップ 2 5 0 1 から始まる処理を実施してキャッシュグループの構成変更方法を求める。その内容を J o b キャッシュ量情報 6 5 0 や J o b リソースマッピング情報に反映させ、求めた方法に従って記憶装置 4 0 に対して変更指示を出す。

## 【 0 2 5 5 】

続いて、J o b の実行時間（開始・終了時刻）や処理件数等の実行情報を J o b 実行履歴として出力する（ステップ 1 3 0 8）。

## 【 0 2 5 6 】

その後、処理を終了する（1 3 0 9）。

## 【 0 2 5 7 】

なお、処理開始時にバッチ J o b 管理情報 7 0 0 中の実行最大時間 7 1 0 を変更した場合には、別途保存した元の値に戻す。また、バッチ J o b 管理情報 7 0 0 で管理された J o b の処理完了時には、他のバッチ J o b 管理情報 7 0 0 中の実行条件 7 0 8 を参照し、完了した J o b に続いて実行すべき J o b が存在するか確認し、その J o b の実行のため本手順を再度実施する。

【 0 2 5 8 】

(IV-5) 記憶装置の諸元の設定変更処理

次に、図 2 3 ないし図 2 7 を用いて記憶装置 4 0 の諸元を変更し、設定する処理について説明する。

【 0 2 5 9 】

まず、図 2 3 を用いてシステム管理プログラム 1 4 0 が J o b を開始する際にキャッシュ量の割当を決定する処理について説明する。

図 2 3 は、システム管理プログラム 1 4 0 が J o b を開始する際にキャッシュ量の割当を決定する処理のフローチャートである。

【 0 2 6 0 】

このキャッシュ量の割当を決定する処理は、システム管理プログラム 1 4 0 が J o b を開始する際、図 2 2 のステップ 1 3 0 4 からコールされるものである。

【 0 2 6 1 】

処理開始時に、上位ルーチンより、これから実行される J o b の J o b 管理情報と、その J o b の実行開始に際して設定されるキャッシュグループ構成案が与えられる（ステップ 2 3 0 1）。

【 0 2 6 2 】

まず、J o b リソースマッピング情報から J o b が利用するデータとそのデータが記憶される記憶装置 4 0 を求める（ステップ 2 3 0 2）。

【 0 2 6 3 】

そして、J o b 管理情報中の I / O 処理内容情報 7 4 0 にあるストレージヒント 7 5 4 からそれぞれのキャッシュ要求量を把握し、システム管理情報 1 4 2 の中の J o b リソースマッピング情報を参照して、L U 2 0 8 毎のキャッシュ要求量を求める。一つのデータが複数の L U 2 0 8 に記憶される場合には、マッピング集約情報からそれぞれの L U 2 0 8 においてデータが記憶される領域量を確認し、それに比例してキャッシュ量を L U 2 0 8 間で分割するものとする。

【 0 2 6 4 】

また、一つの L U 2 0 8 に複数のデータが記憶されている場合には、その総和が L U 2 0 8 のキャッシュ要求量であるとする。その値を処理開始時に与えられ

た構成に従ってキャッシュグループ毎に集計し、J o b の初期設定として J o b キャッシュ量情報 6 5 0 に記憶する。さらに、その値を記憶装置 4 0 毎に集計し、キャッシュ要求量とする。

#### 【0265】

続いて、各記憶装置 4 0 毎に、ステップ 2 3 0 2 で求めたキャッシュ要求量をフリーキャッシュグループから割当可能か確認する。フリーキャッシュグループのキャッシュ量からキャッシュ要求量を差し引いた量が、事前に定められている閾値を下回らない場合には、割当可能とする。割当不可能と判断された場合には以下の処理をおこなう。

#### 【0266】

まず、キャッシュ割り当て中の J o b に与えられた J o b 優先度 7 0 4 を J o b 管理情報から把握する。そして、その記憶装置 4 0 のキャッシュグループに関して、それを利用している J o b を J o b リソースマッピング情報から把握し、その J o b の J o b 優先度 7 0 4 を求め、その値がキャッシュ割り当て中の J o b より低いものを求める。そのような J o b が存在しない場合には、現在の割当量から変更なしとする。存在する場合には、それらのキャッシュ容量の一部をキャッシュ割当を要求している J o b へ再割当する方法を決定する（ステップ 2 3 0 3）。

#### 【0267】

例えば、低優先度の J o b に割り当てられているキャッシュを、その量に比例した分をキャッシュ割り当て中の J o b に再割り当てする。このとき、キャッシュ量が削減された J o b に対して、J o b キャッシュ量情報 6 5 0 中の割当情報 6 5 2 に、設定量を削減後の値、設定量を“キャッシュ不足”として新世代のデータを設定する。

#### 【0268】

続いて、ステップ 2 3 0 3 で求めた割当量に従って、キャッシュグループへの割当方法を決定する（ステップ 2 3 0 4）。

#### 【0269】

フリーキャッシュグループから要求量を取得できたときには、その値がそのま



ま割当量になる。取得できなかったときは、総和が割当量になるようにキャッシュグループ毎に要求量に比例した量を割り当てる。このとき、Jobキャッシュ量情報650中の割当情報652に、設定量として実際に割り当てられた量であり設定理由が“キャッシュ不足”である新世代のデータを設定する。

## 【0270】

そして、処理を終了する（ステップ2305）。なお、キャッシュ量を削減されたJobに対して、後に図25で説明するキャッシュチューニング処理を実施してもよい。

## 【0271】

次に、図24を用いてシステム管理プログラム140がJobの終了時処理として実施するキャッシュの再割当処理について説明する。

図24は、システム管理プログラム140がJobの終了時処理として実施するキャッシュの再割当処理のフローチャートである。

## 【0272】

このキャッシュ量の再割当処理は、システム管理プログラム140がJobを終了する際、図22のステップ1307からコールされるものである。

## 【0273】

この処理では、処理開始時に、上位ルーチンから終了したJobのJobID622が与えられる（ステップ2501）。

## 【0274】

まず、Jobキャッシュ量情報650を参照し、終了するJobが利用していたキャッシュ量を開放キャッシュ量として把握し、各記憶装置40毎にその量を集計する（ステップ2502）。

## 【0275】

そして、対応エントリをJobキャッシュ量情報650から削除する。さらに、Jobリソースマッピング情報を参照して、終了するJobのデータのみが記憶されるLU208のみが属するキャッシュグループを把握し、そのキャッシュグループは削除するものとする。

## 【0276】

続いて、J o b キャッシュ量情報 6 5 0 を参照して、終了する J o b がデータを記憶していた記憶装置 4 0 において、要求値分のキャッシュ量が割り当てられていない J o b を把握する。対応する割当情報 6 5 2 の設定理由が“キャッシュ不足”のエントリのものがこれに相当する。そして、それらの J o b の J o b 優先度 7 0 4 を J o b 管理情報から把握し、J o b 優先度 7 0 4 が高いものから順にキャッシュ量の要求値からの不足分を充足するように開放キャッシュの再割当をおこなう（ステップ 2 5 0 3）。

## 【 0 2 7 7 】

ステップ 2 5 0 3 において、全ての J o b に対してキャッシュの再割当を行った後も開放キャッシュに残りが存在する場合には、それらはフリーキャッシュグループに割り当てるものとする（ステップ 2 5 0 4）。

## 【 0 2 7 8 】

そして、処理を終了する（ステップ 2 5 0 5）。なお、不足分が充足された J o b のキャッシュグループに関しては、J o b キャッシュ量情報 6 5 0 の対応エントリを不足前の世代に戻す。

## 【 0 2 7 9 】

次に、図 2 5 を用いてモニタ情報を基にしたオンライン J o b 向けにキャッシュ量をチューニングする処理について説明する。

図 2 5 は、モニタ情報を基にしたオンライン J o b 向けにキャッシュ量をチューニングする処理のフローチャートである。

## 【 0 2 8 0 】

この処理は、図 2 0、図 2 1 の J o b のチューニング処理のステップ 1 5 0 7、ステップ 1 5 1 9 からコールされる処理である。

## 【 0 2 8 1 】

この処理では、処理開始時に上位ルーチンからチューニングを実施する J o b の J o b I D 6 2 2 と、必要に応じてキャッシュ割当量を削減してはならないキャッシュグループが与えられる（ステップ 2 7 0 1）。

## 【 0 2 8 2 】

まず、指定された J o b が利用するデータを保持する記憶装置 4 0、L U 2 0

8、キャッシュグループをJobリソースマッピング情報から求める（ステップ2702）。

【0283】

続いて、キャッシュ量を変更した際のキャッシュヒット率の変化量を計算する（ステップ2703）。

【0284】

本実施形態においては、記憶装置40における実領域分割数502と仮想管理領域数504の値がそれぞれRとVであり、LU208が属するキャッシュグループの現在のキャッシュ量をCとしたとき、キャッシュモニタ情報362中のキャッシュヒット回数累計370における第i領域（ $0 < i \leq R$ ）のヒット数は、そのLU208が属するキャッシュグループのキャッシュ量を $iC/R$ から（ $i-1$ ） $C/R$ に削減したときにヒットしなくなる回数、第i領域（ $R < i \leq R+V$ ）のヒット数は、キャッシュ量を（ $i-1$ ） $C/R$ から $iC/R$ に増加させたときにヒットするようになる回数と考えられる。

【0285】

ここで、各LU208毎のリードアクセス実行時の第i領域（ $0 < i \leq R+V$ ）での平均ヒット率は、図12に示されるモニタ履歴情報510に、記憶装置40における実領域分割数502と仮想管理領域数504は別途システム管理情報142中に保持されている。例えば、キャッシュ量をCから $C/R$ 減らしたときの平均実ヒット率の削減分が第R領域における平均ヒット率であるから、LU208におけるキャッシュ量が（ $i-1$ ） $C/R$ から $iC/R$ の範囲（ $0 < i \leq R$ ）での単位キャッシュ量あたりの平均ヒット率削減分は（第i領域での平均ヒット率） $\times R/C$ として近似可能である。同様に、キャッシュ量が（ $i-1$ ） $C/R$ から $iC/R$ の範囲（ $R < i \leq R+V$ ）での単位キャッシュ量あたりの平均ヒット率増加分は（第i領域での平均ヒット率） $\times R/C$ として近似可能である。

【0286】

この値をJobが利用するデータを保持するLU208について全て計算し、ヒット率変化率とする。なお、以下のステップを含めてモニタ履歴情報510か

らは最も直近のサンプル524による値、あるいは、事前に定められた直近のある期間の平均値を取得するものとする。

#### 【0287】

次に、システム管理情報412の中のJ o bリソースマッピング情報を用いて、J o bが利用するデータを保存するLU208とそれに対応するHDD16を把握する（ステップ2704）。

#### 【0288】

そして、HDD性能情報612のHDD16のアクセス性能情報614を参照してHDD16のリードヒット時の平均応答時間とリードミス時の平均応答時間を求め、（リードミス時の平均応答時間）－（リードヒット時の平均応答時間）を計算しLU208の応答時間変化量とする。なお、LU208中のデータが性能の異なるHDD16に分割されて記憶される場合には、HDD16毎に応答時間変化量を求め、HDD16に記憶されるデータ量で重み付け平均を計算した値をLU208の応答時間変化量とする。

#### 【0289】

記憶装置40毎に、

$$I = \Sigma (\text{LU処理時間変化の平均})$$

ただし、 $(\text{LU処理時間変化の平均}) = (\text{応答時間変化量}) \times (\text{LUの平均リードI/O回数}) \times (\text{ヒット率変化量})$

として、Iを最大化するようなキャッシュ割当の変更方法を求める（ステップ2705）。

#### 【0290】

ここで、 $\Sigma$ は、ある記憶装置40における指定されたJ o bのデータを保持するLU208に関して総和を取ることを意味する。また、応答時間変化量は、ステップ2704で求めたものを用い、LU208の平均リードI/O回数はモニタ履歴情報510から取得する。ヒット率変化量はステップ2703で求めたヒット率変化率とキャッシュ割当の変更量から求めることができる。このとき、ヒット率変化率は、キャッシュグループのキャッシュ量により変化することに注意する。

## 【0291】

最大化には、例えば、以下のアルゴリズムを用いる。

## 【0292】

まず、あるキャッシュグループに対して、単位キャッシュ量を増やした場合と減らした場合の I を計算する。そして、キャッシュ量を減らした場合に最も I が大きい（影響が小さい）キャッシュグループからキャッシュ量を増やした場合に最も I の値が大きい（効果が高い）キャッシュグループへ単位量分キャッシュ割当を変更したとした場合の I の値を再計算し、それがある閾値以上のときに「効果あり」としてキャッシュ割当の変更をおこなうとする。以下、キャッシュ割当の変更がおこなわれたとして上記確認を繰り返し実施し、さらに割当変更をおこなうことに効果なしと判断されるか、あるいは、事前に定められる一定量（例えば、記憶装置 40 で Job が利用しているキャッシュ量の一定割合）分の割当変更をおこなった場合に、この確認の繰り返しを終了させる。なお、処理開始時にキャッシュ割当量を削減してはならないキャッシュグループが指定された場合には、その制約を守るようにする。

## 【0293】

そして、処理を終了する（ステップ 2706）。このとき、図 16 の Job リソース管理情報 620 中の Job キャッシュ量情報 650 中の割当情報 652 の値を更新する。

## 【0294】

次に、図 26 を用いて I/O 処理性能の変更方法を決定する処理について説明する。

図 26 は、I/O 処理性能の変更方法を決定する処理のフローチャートである。

## 【0295】

この処理は、Job とポートの割当てを調整して変更する処理であり、図 22 の Job を実行する処理のステップ 1303、図 20、図 21 の Job のチューニング処理のステップ 1515、1517 からコールされる処理である。

## 【0296】

また、この処理は、処理開始時に、上位ルーチンから J o b を新規に追加するときや J o b の完了により割当を開放した際の変更をするときには、その J o b の識別子と追加・開放を示す情報が与えられ、既に割り当てられた I / O 処理量を変更する場合には、変更する J o b の識別子と割当を変更したいポート 2 6 の識別子、そのポート 2 6 へ新たに設定したい性能割合 6 3 2 が与えられる（ステップ 1 6 0 1）。他の J o b が要求する I / O 処理量は、特に指定がない限り、J o b ポート設定情報 6 3 0 とそこから識別される J o b の J o b 管理情報 7 4 0 から把握する。

## 【 0 2 9 7 】

まず、割当変更の対象となる記憶装置 4 0 における I / O パス I / F 3 2 のポート 2 6 を把握する（ステップ 1 6 0 2）。

## 【 0 2 9 8 】

J o b の新規追加・削除の時には、J o b の識別子から J o b リソースマッピング情報を参照し、その J o b が利用するポート 2 6 を全て把握し、それらを処理量割当の変更対象とする。ポート 2 6 の I / O 処理量が直接指定されたときは、処理量割当の変更対象は指定されたポート 2 6 のみとする。その後、求めたポート 2 6 を利用する現在実行中の J o b を、システム管理情報 4 1 2 中の J o b リソースマッピング情報を用いて全て把握する。

## 【 0 2 9 9 】

次に、ステップ 1 6 0 2 で求めた各ポート 2 6 において、各 J o b の要求値通り I / O 処理量を割り当てるとした割当案を作成する（ステップ 1 6 0 3）。

## 【 0 3 0 0 】

ここで要求値とは、処理が呼び出されたときに性能割合 6 3 2 が指定されたときにはその値であり、そうでない場合には以下のように求める。

## 【 0 3 0 1 】

まず、J o b リソースマッピング情報から求まるポート 2 6 を通して J o b がアクセスする各データに対して、その J o b の J o b 管理情報中の I / O 処理内容情報 7 4 0 を参照し、そのデータに対応するアクセス順 7 4 2 の値によりグループ化する。次に、各グループ毎にそれに属するデータの I / O ベース性能 7 4

8のそのポート26における性能値を性能割合632に変換した値の和を計算する。それぞれのグループにおける計算値とJobポート設定情報630中の現在の性能割合632を比較し、その中で最も大きな値を要求値とする。

【0302】

ステップ1603で求めた割当案が実現可能か、すなわち、各ポート26における各Jobからの要求値の和が100%を超えないかを確認する（ステップ1604）。

【0303】

実現不可能なポートが存在する場合には、ステップ1605に進む。全てのポートで実現可能である場合には、ステップ1610に進み、現在の割当案をI/O処理量の割当方法とし、処理を終了する。

【0304】

ステップ1604で全てのポートで実現可能である場合と判断された場合、ステップ1604の確認で問題が発生するとされたポート26に対してのみ割当案の再計算をおこなう。

【0305】

まず、オンライン業務向けJobの負荷確認に用いるモニタ情報の参照時間を定める（ステップ1605）。

【0306】

基本的には、現時刻から、事前に与えられた時間（例えば1時間）をその時間とする。ただし、ステップ1602で求めたJobに関して、あるJobがバッチJob管理情報700で管理される場合には、そのJobが開始された時間から最大実行時間710の値で与えられる時間、処理が実行されるとして実行終了時刻を見積り、その終了時刻が前述の参照期間の終了時刻より早い場合には、そのJobの実行終了見積時刻をモニタ情報の参照時間の終了時刻としてもよい。

【0307】

次に、要求性能に影響を与えないと見積られる範囲でI/O処理量を削減した割当案を作成する（ステップ1606）。

【0308】

割当値は、処理が呼び出されたときに処理量が指定されものはその値とする。

#### 【0309】

バッチ J o b 管理情報 700 で管理される J o b に対しては次のように計算する。まず、割当量が定まっていないポート 26 における処理性能は、要求値に J o b 毎に定まる補正係数を掛けた値となるとし、そのときに処理実行時間見積値が最大実行時間 710 で与えられる時間となる補正係数を計算する。前述の J o b の処理実行時間見積方法の説明に従うことにより補正係数を求める単純な方程式が得られ、これを解くことによりその値を求める。バッチ J o b 管理情報 700 中の J o b 優先度 704 を用いて優先度制御情報 880 を参照して求める余裕係数 882 と計算した補正係数を要求値に乗じた値と要求値を比較し、小さな値の方を割当値とする。

#### 【0310】

オンライン J o b 管理情報 770 で管理される J o b に対しては、次のように求める。まず、割当量が定まっていないポート 26 を経由してアクセスされるデータを記憶する L U 208 を J o b リソースマッピング情報から求める。過去のその L U 208 への対象ポート 26 を介しての I / O 処理量をモニタ履歴情報 510 から把握する。その際に参照するデータは、事前に定められた期間（例えば 1 週間）以内のステップ 1605 で求めたモニタ情報の参照時間内のモニタ値と、直前のモニタ値とする。そのモニタ値から計算する性能割合 632 の最大値にオンライン J o b 管理情報 770 中の J o b 優先度 704 から求まる余裕係数 882 を乗じた値と要求値を比較し、小さな方を割当値とする。

#### 【0311】

そして、求めた各ポート 26 における各 J o b からの割当値の和が 100% を超えないか確認する（ステップ 1607）。なお、和が 100% を超えないポート 26 に関しては、総和が 100% になるように割当値の修正をおこなう。この修正は、J o b 優先度 704 が高い J o b から順に要求値からの差分を順次割り当てることによりおこなう。

#### 【0312】

一つ以上のポート 26 で割当値の和が 100% を超える問題が解消されない場



合にステップ1608に進む。全てのポートで問題が解消された場合にはステップ1610に進み、現在の割当案をI/O処理量の割当方法とし、処理を終了する。

## 【0313】

以降、ステップ1607までで、問題が解決されなかったポート26に対して、Jobの優先度を考慮してI/O処理量を割り当てる処理をおこなう（ステップ1608）。

## 【0314】

例えば、割当が決まっていないポート26毎に、そのポート26への各Jobの割当量は、ステップ1606で求めた割当量にポート26毎に定める新たな補正係数を（Jobに与えられたJob優先度704の値）乗じたものとし、その総和が100%になるように補正係数を決定、その値を各Jobへの割当値とする。

## 【0315】

ステップ1608でI/O処理量の割当案が作成されたポート26を介してデータをアクセスするJobに関して、バッチJob管理情報700で管理されるJobにおいては割当案による処理時間見積り値が最大実行時間710を満たさない場合、オンラインJob管理情報770で管理されるJobにおいてはI/O処理量のステップ1606で求めた割当値を確保できなかったポート26が存在する場合に、要求性能を満たさない可能性が高い旨の情報を出力する（ステップ1609）。

## 【0316】

これは、管理端末110の画面114にその旨を表示してもよいし、また、システム管理プログラム140が各種イベントの履歴を記憶するログに追加してもよい。

## 【0317】

そして、処理を終了する（ステップ1610）。

## 【0318】

次に、図27を用いてモニタ情報を基にしたデータ移動によるHDD過負荷状

態の解消方法を決定する処理について説明する。

図27は、モニタ情報を基にしたデータ移動によるHDD過負荷状態の解消方法を決定する処理のフローチャートである。

【0319】

この処理は、図20のJobのチューニング処理のステップ1508からコールされる処理である。

【0320】

処理開始時に、上位ルーチンからデータの移動元となる移動元HDD16を有する記憶装置40と移動元HDD16とそこに記憶されているデータを移動するLU208のそれぞれの識別子が与えられる（ステップ2901）。

【0321】

まず、各仮想化機構から領域マッピング情報300の最新状態を取得し、マッピング集約情報を最新の状態に更新する。その後、各仮想化機構の領域マッピング情報300におけるフリーエントリ314に対応する下位管理構造対応領域情報304を基に「フリー領域」として把握し、可能であれば、仮想化スイッチ60、ボリュームマネージャ78におけるフリー領域を開放する（ステップ2902）

ここで、「フリー領域」とは、フリーエントリ314に対応する下位管理構造対応領域情報304のうちで、システム管理情報142中に既に存在する「データ移動要求」のデータの移動先として利用されていないものである。また、データ移動要求とは、このステップ2901から開始される処理が作成するある記憶装置40におけるデータ移動によりLU208とHDD16の対応を変更する方法に関する情報であり、そのデータ移動を実施する記憶装置40の装置ID572と移動元となるHDD16のHDDID394とデータを移動すべきLU208のLUID364、移動先となるHDD16のHDDID394とその中の領域に関する情報を含んでいる。この情報はシステム管理情報142中に保存される。

【0322】

把握したフリー領域のうち、仮想化スイッチ60、ボリュームマネージャ78

におけるフリー領域に関しては、可能であればその領域を開放する指示を仮想化スイッチ 6 0、ボリュームマネージャ 7 8に発行することにより、開放することができる。ある領域が開放された際に、開放された領域を構成していたものが仮想構造であり、かつ、その仮想構造の全ての領域が上位階層で利用されなくなった場合には、その仮想構造を提供する仮想化機構にその仮想構造を削除するよう指示し、その仮想構造を構成していた管理構造の領域をフリー領域化する。以下、再帰的にこの領域開放処理を行い、それらについてもフリー領域の把握をおこなう。

## 【 0 3 2 3 】

移動元 HDD 1 6 を有する記憶装置 4 0 内の HDD 1 6 のうち、移動元 HDD 1 6 上に記憶されている指定された LU 2 0 8 のデータを全て連続したフリー領域へ格納可能な HDD 1 6 を求め、その HDD 1 6 のリードミス時の平均応答時間を HDD 性能情報 6 1 2 中のアクセス性能情報 6 1 4 から求め、その値が移動元 HDD 1 6 の値以下（同等以上のリードミス性能）の HDD 1 6 を選択する（ステップ 2 9 0 3）。なお、HDD 1 6 が選択されない可能性もある。

## 【 0 3 2 4 】

選択された HDD 1 6 において、モニタ履歴情報 5 1 0 中の最近のある一定期間（例えば 1 週間）における HDD 1 6 の稼働率のサンプル 5 2 4 を把握し、その期間内の各サンプル 5 2 4 で、そこがデータの移動先となった場合の、データ移動後の HDD 1 6 全体の稼働率（HDD 1 6 にデータを記憶する LU 2 0 8 における HDD 1 6 の稼働率の総和）を見積る（ステップ 2 9 0 4）。

## 【 0 3 2 5 】

ここで、移動された LU 2 0 8 のデータによる HDD 1 6 の稼働率は、移動先と移動元で HDD 1 6 の稼働率はリードミス性能に比例するとして、移動元 HDD 1 6 における LU 2 0 8 分の稼働率を補正した値として見積る。

## 【 0 3 2 6 】

各サンプル 5 2 4 の見積り値のうち、最も値の大きいものをその HDD 1 6 のデータ移動後の稼働率とし、その値が最も小さな HDD 1 6 を選択し、それが事前に定められる閾値（例えば 5 0 %）未満か否かを確認する。閾値未満の場合に

はそこへ移動元HDD16におけるLU208のデータを移動するデータ移動要求を作成する（ステップ2905）。閾値以上の場合、もしくは、ステップ2903でHDD16が一つも選択されなかった場合には、作成失敗としてデータ移動要求は作成しない。

【0327】

そして処理を終了する（ステップ2906）。

【0328】

〔実施形態2〕

以下、本発明に係る第二の実施形態を、図28を用いて説明する。

図28は、本発明の第二の実施形態に係る記憶装置の性能を管理する計算機システムの構成図である。

【0329】

第一の実施形態では、サーバ70と記憶装置40の間に仮想化スイッチ60が置かれ、I/Oパス34と、ネットワーク24により接続されていた。そして、図2に示すような管理構造により、データが仮想化されていた。

【0330】

本実施形態では、記憶装置40bは、ファイルを外部装置に対して提供し、そのファイルは、ネットワークファイルシステムプロトコルを用いてネットワーク経由でアクセスされる。

【0331】

以下、第一の実施の形態との差異に関して説明する

本実施形態においては、I/Oパス34、仮想化スイッチ60が存在しない。ネットワークI/F24のポート26には、システム内で一意に識別可能なポートID384が付加される。

【0332】

サーバ70は、I/OパスI/F32を有せず、ネットワークI/F22のみ有している。OS72は、外部装置が提供するファイル202をネットワークファイルシステムプロトコルを用いて、ネットワーク24経由でアクセスするネットワークファイルシステム82を含む。

## 【 0 3 3 3 】

ネットワークファイルシステム 8 2 は、領域マッピング情報 3 0 0 を OS 管理情報 7 4 中に有する。また、DBMS 9 0 や AP プログラム 1 0 0 により認識されるファイル 2 0 2 と記憶装置 4 0 b から提供されるファイル 2 0 2 があるルールに従って対応する場合、その対応関係を定める情報のみが OS 管理情報 7 4 中に保持されても良い。この場合には、システム管理プログラム 1 4 0 は対応関係を定める情報を取得し、それから領域マッピング情報 3 0 0 を作成し、マッピング集約情報中に記憶する。

## 【 0 3 3 4 】

記憶装置 4 0 b は、I/O パス I/F 3 2 を有せず「ファイル」という概念で、外部装置に対してデータのアクセスをさせるものである。記憶装置 4 0 b の制御プログラム 4 4 b に変更され、次の機能を有する。制御プログラム 4 4 b は、第一の実施形態のファイルシステム 8 0 が提供する機能を有し、記憶装置 4 0 b 内に存在する LU 2 0 8 の記憶領域を仮想化して、ファイル 2 0 2 という概念でデータアクセスを可能にする。また、制御プログラム 4 4 b は一つ以上のネットワークファイルシステムプロトコルを解釈し、ネットワーク 2 4、ネットワーク I/F 2 2 経由で外部装置からそのプロトコルを用いて要求されるファイルアクセス処理を実施する。

## 【 0 3 3 5 】

データのマッピングに関しては、データのマッピング階層構成において、ファイル 2 0 2 以下が全て記憶装置 4 0 b により提供されるようになり、サーバ 7 0 は OS 7 2 内のネットワークファイルシステム 8 2 を用いて記憶装置 4 0 b 上にあるファイル 2 0 2 をアクセスする。

## 【 0 3 3 6 】

記憶装置内の論理階層としては、図 2 のようなファイル 2 0 2 と LU 2 0 8 間に、論理ボリューム 2 0 4、仮想ボリューム 2 0 6 を含んだものとしてもよいし、単に、ファイル 2 0 2 - LU 2 0 8 - HDD 1 6 という、より簡易な構成でもよい。

## 【 0 3 3 7 】

その他の差異をまとめると以下ようになる。I/Oパス I/F 3 2 のポート 2 6 をネットワーク I/F 2 2 のポート 2 6 に対応させる。記憶装置 4 0 が有していた優先アクセス制御機能に関してもこれを当てはめる。

【 0 3 3 8 】

記憶装置 4 0 の L U 2 0 8 に対応するものとして、記憶装置 4 0 b のファイル 2 0 2 を対応させる。キャッシュグループのメンバの管理単位もファイル 2 0 2 が単位となる。

【 0 3 3 9 】

モニタ情報の取得や、各種処理においては、基本的に上記の変更をおこなうことにより、第一の実施の形態のものをそのまま当てはめることができる。

【 0 3 4 0 】

【発明の効果】

本発明によれば、DBMS が稼動する計算機システムにおいて、記憶装置に対する性能管理を、ユーザの業務における性能指標を用いておこなえるようにし、性能管理を容易化することができる。

【 0 3 4 1 】

また、本発明によれば、DBSM が稼動する計算機システムにおいて、記憶装置の性能チューニング処理を自動化し、性能管理コストを削減することができる。

【図面の簡単な説明】

【図 1】

本発明の第一の実施形態に係る記憶装置の性能を管理する計算機システムの構成図である。

【図 2】

本発明の第一の実施形態に係る DBMS 9 0 が管理するデータのデータマッピングの階層構成図である。

【図 3】

領域マッピング情報 3 0 0 のデータ構造を示す模式図である。

【図 4】

上位ポートマッピング情報 3 2 2 のデータ構造を示す模式図である。

【図 5】

データ記憶領域情報 3 4 2 のデータ構造を示す模式図である。

【図 6】

キャッシュグループ情報 4 6 0 のデータ構造を示す模式図である。

【図 7】

ストレージモニタ情報 3 6 0 のデータ構造を示す模式図である。

【図 8】

ポートスループット設定情報 4 4 0 のデータ構造を示す模式図である。

【図 9】

ポートモニタ情報 4 0 0 のデータ構造を示す模式図である。

【図 1 0】

オンライン J o b モニタ情報 4 3 0 のデータ構造を示す模式図である。

【図 1 1】

ストレージ性能情報 6 1 0 のデータ構造を示す模式図である。

【図 1 2】

モニタ履歴情報 5 1 0 のデータ構造を示す模式図である。

【図 1 3】

バッチ J o b 管理情報 7 0 0 のデータ構造を示す模式図である。

【図 1 4】

バッチ S Q L 設計情報 8 5 0 のデータ構造を示す模式図である。

【図 1 5】

オンライン J o b 管理情報 7 7 0 のデータ構造を示す模式図である。

【図 1 6】

J o b リソース管理情報 6 2 0 のデータ構造を示す模式図である。

【図 1 7】

優先度制御情報 8 8 0 のデータ構造を示す模式図である。

【図 1 8】

J o b 管理情報 7 0 0 中の C P U 処理時間見積方法 7 1 6 と I / O 処理内容情

報 7 4 0 a をモニタ情報の実測により設定する処理のフローチャートである。

【図 1 9】

システム管理プログラムのモニタ情報の収集とそのモニタ情報による J o b チューニング処理のゼネラルチャートである。

【図 2 0】

モニタ情報による J o b チューニング処理のフローチャートである（その一）

【図 2 1】

モニタ情報による J o b チューニング処理のフローチャートである（その二）

【図 2 2】

システム管理プログラムによる J o b 実行処理のフローチャートである。

【図 2 3】

システム管理プログラム 1 4 0 が J o b を開始する際にキャッシュ量の割当を決定する処理のフローチャートである。

【図 2 4】

システム管理プログラム 1 4 0 が J o b の終了時処理として実施するキャッシュの再割当処理のフローチャートである。

【図 2 5】

モニタ情報を基にしたオンライン J o b 向けにキャッシュ量をチューニングする処理のフローチャートである。

【図 2 6】

I / O 処理性能の変更方法を決定する処理のフローチャートである。

【図 2 7】

モニタ情報を基にしたデータ移動による H D D 過負荷状態の解消方法を決定する処理のフローチャートである。

【図 2 8】

本発明の第二の実施形態に係る記憶装置の性能を管理する計算機システムの構成図である。



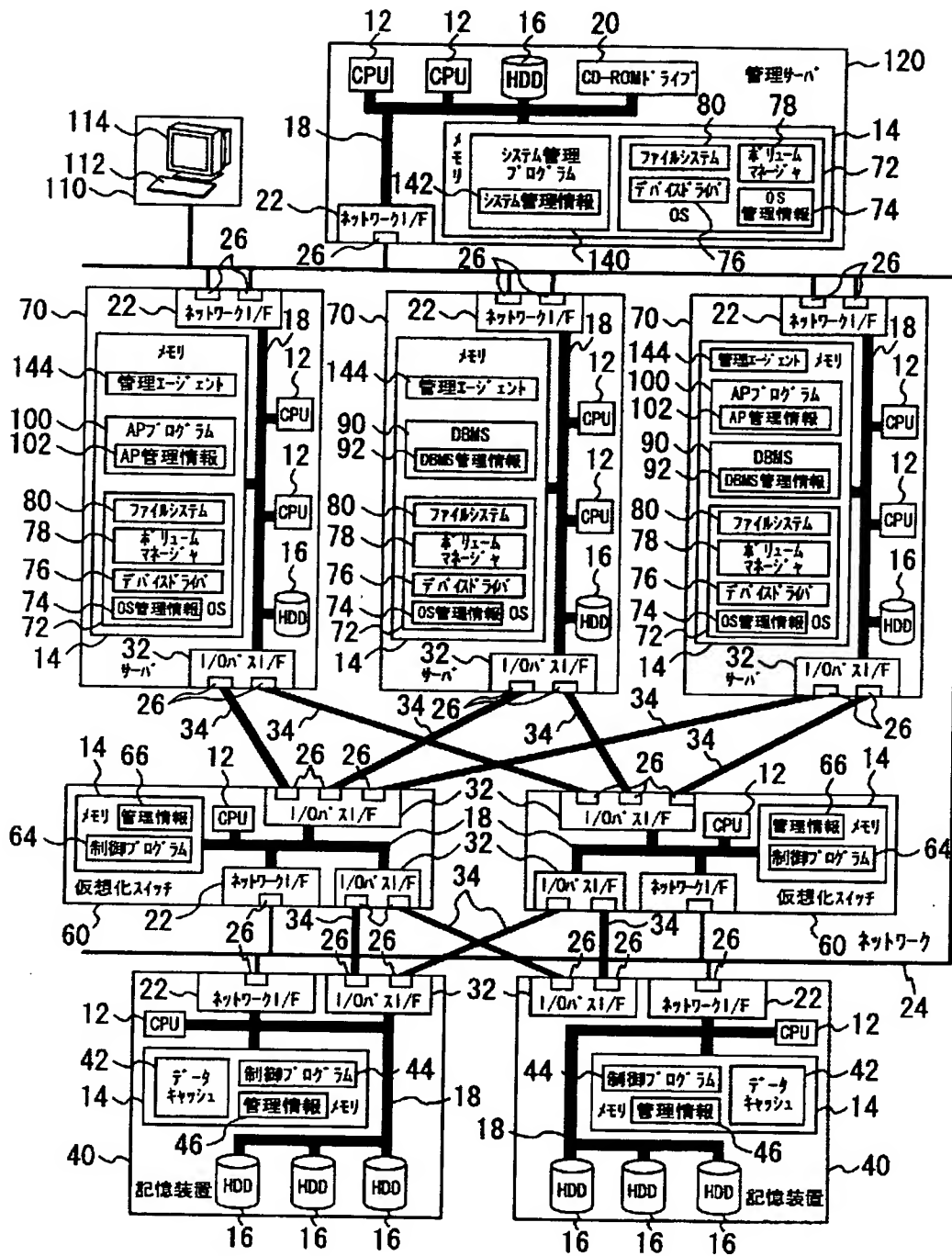
【符号の説明】

1 6 … HDD、2 2 … ネットワーク I / F、2 4 … ネットワーク、2 6 … ポート、3 2 … I / O パス I / F、3 4 … I / O パス、4 0 … 記憶装置、6 0 … 仮想化スイッチ、7 0 … サーバ、9 0 … DBMS、1 0 0 … AP プログラム、1 2 0 … 管理サーバ、1 4 0 … システム管理プログラム。

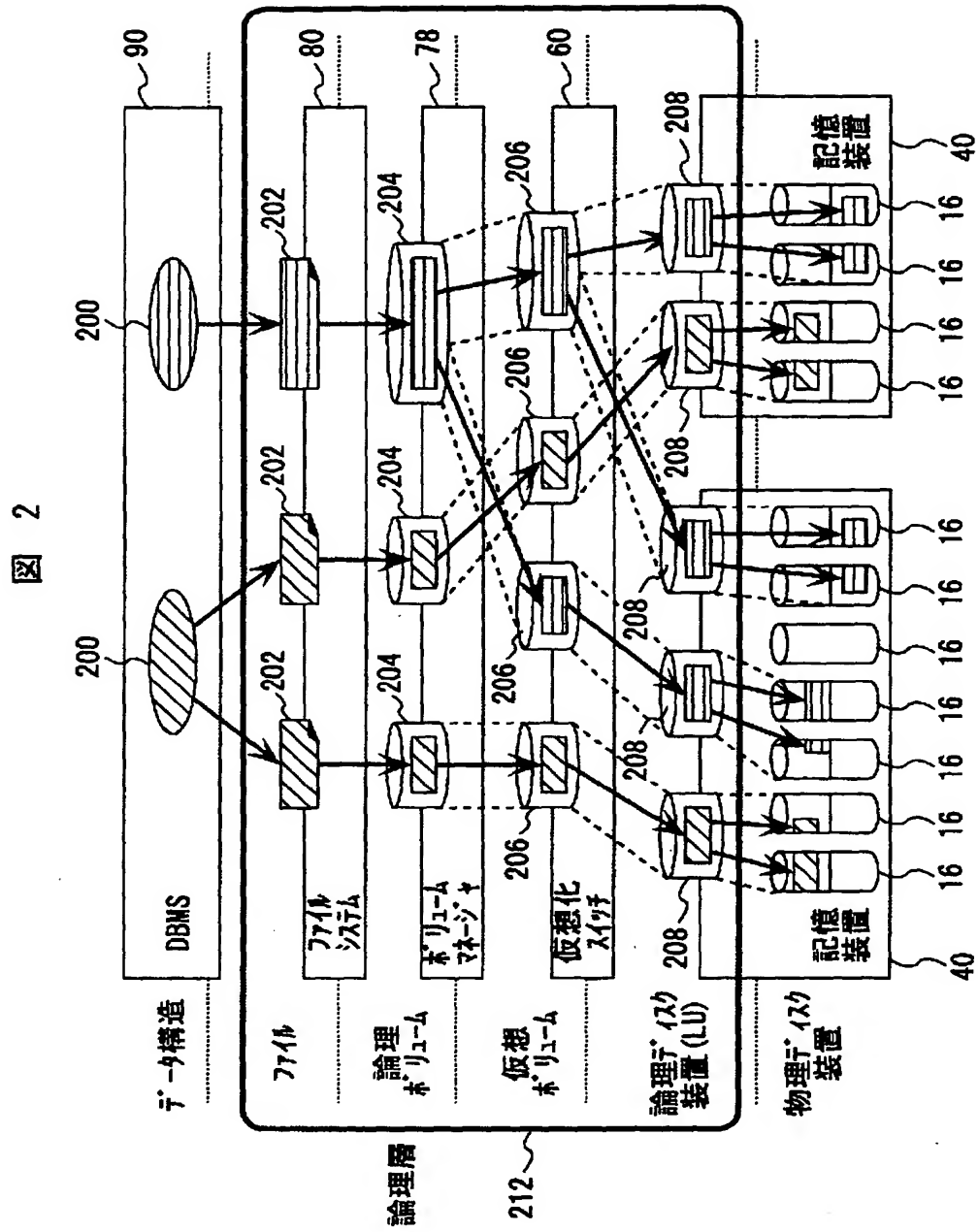
【書類名】 図面

【図 1】

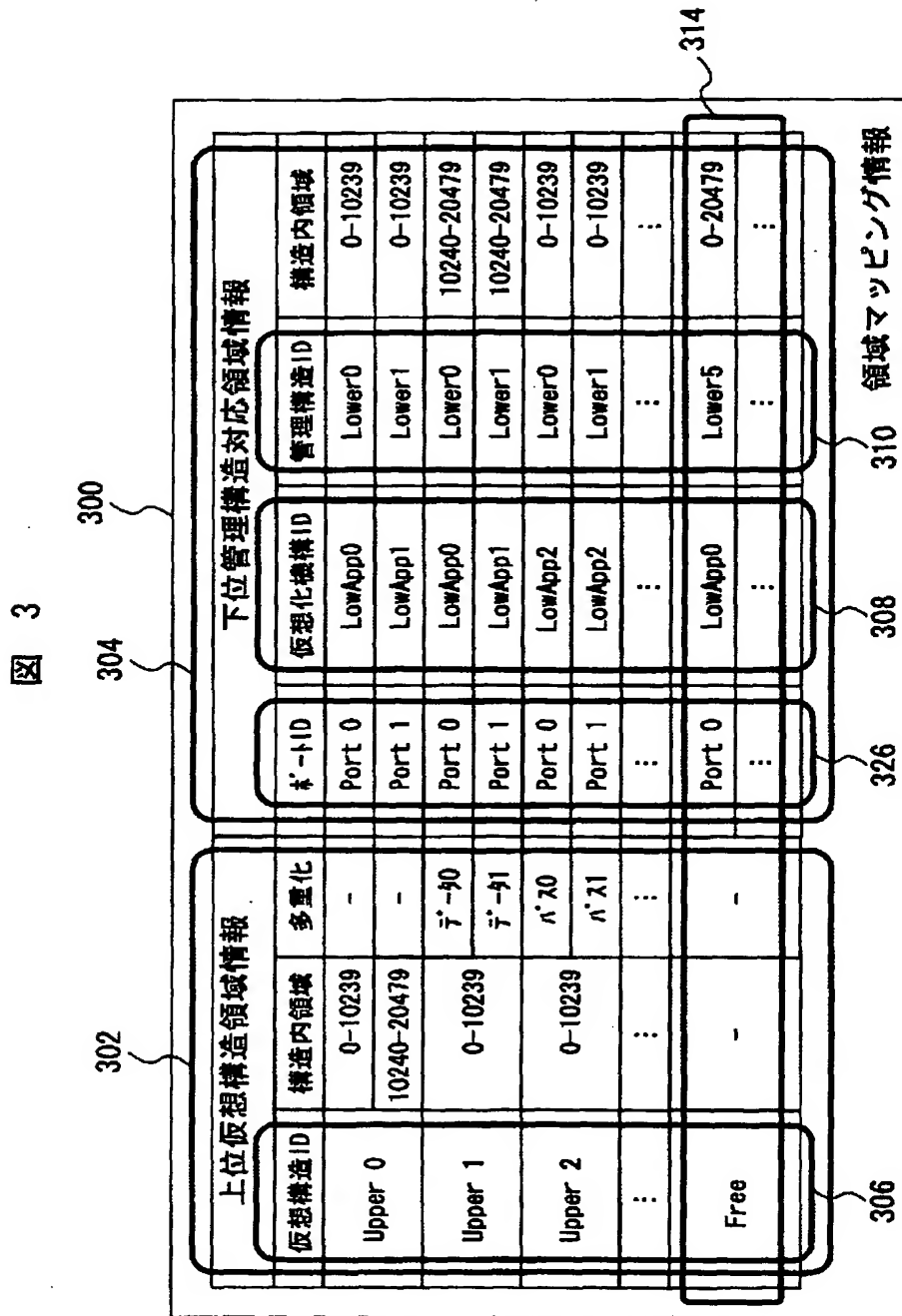
図 1



【図2】

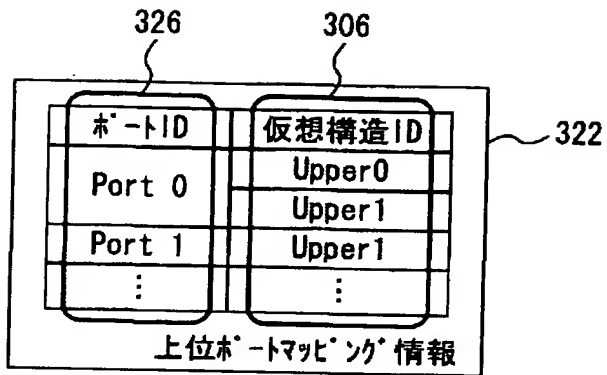


【図 3】



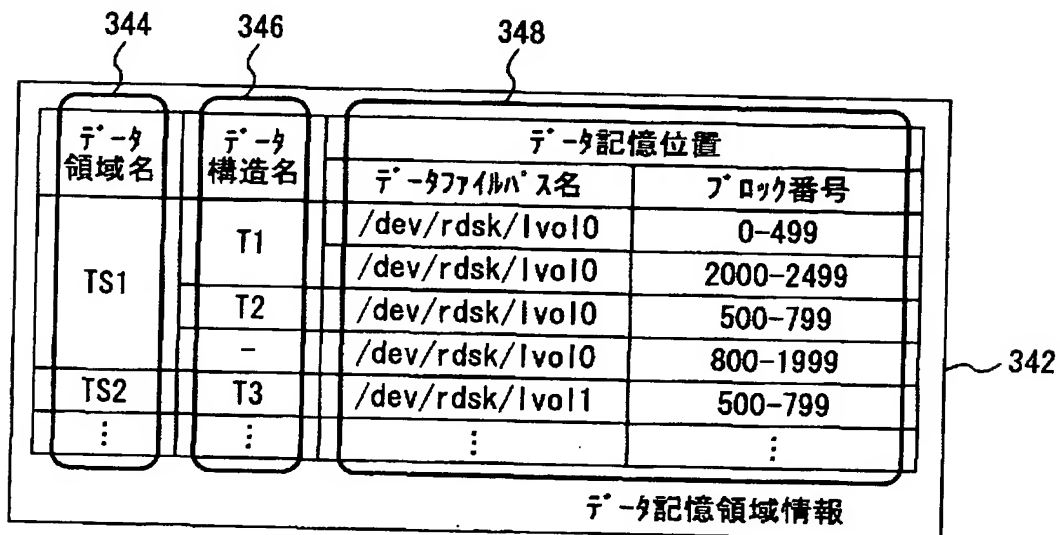
【図 4】

図 4



【図 5】

図 5



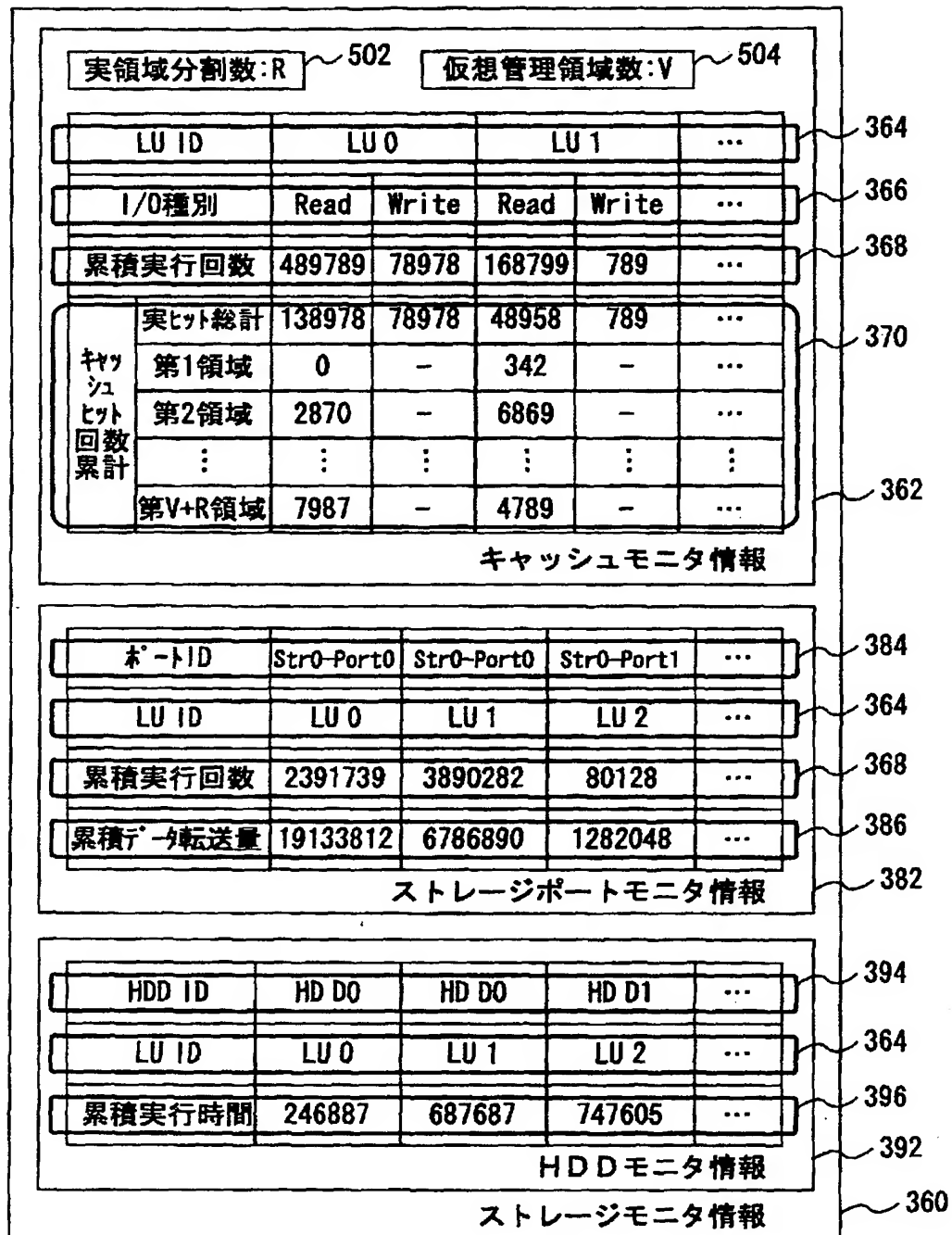
【図 6】

図 6

キャッシュ・グループ ID	CGrp0	CGrp1	...	462
割当セグメント数	10240	20480	...	464
LU ID	LU0, LU2, LU3, ...	LU 1	...	364
キャッシュ・グループ 情報				460

【図 7】

図 7



【図 8】

図 8

ホストID	Str0-Port0	Str0-Port2	...	384
アクセス元ホストID	Host0-Port1	Host1-Port0	...	442
最大データ転送量	1 (MB/s)	10 (MB/s)	...	444
最大I/O処理量	300	500	...	446
ホストスループット設定情報				440

【図 9】

図 9

ホストID	Host0-Port0		Host0-Port1		...	384
I/O種別	Read	Write	Read	Write	...	366
累積実行回数	48978	7897	16879	570	...	368
累積応答時間	138978	6318	38958	456	...	402
累積データ転送量	391824	63176	113899	3756	...	386
ホストモニタ情報						400



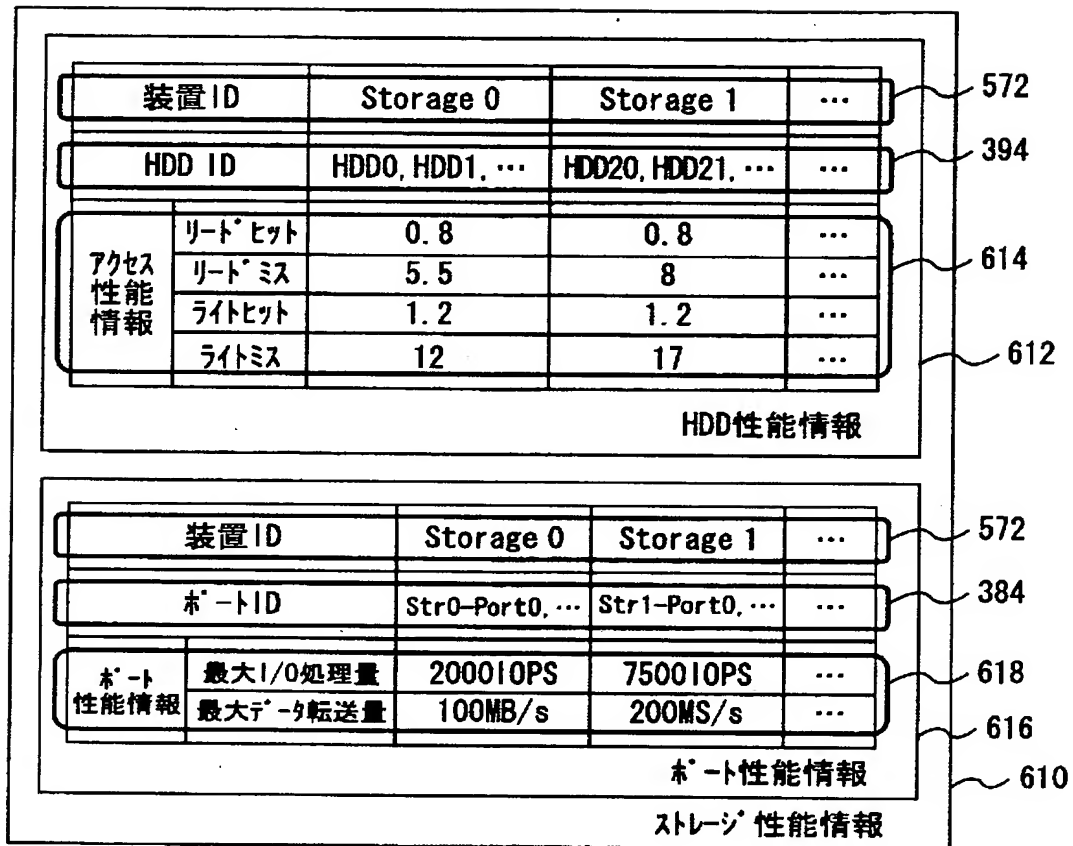
【図 1 0】

図 10

処理ID	1	2	...	432
累積実行回数	48978	7897	...	368
累積応答時間	138978	26318	...	402
累積処理時間	102956	24637	...	396
オンラインJobモニタ情報				430

【図 11】

図 11



【図 12】

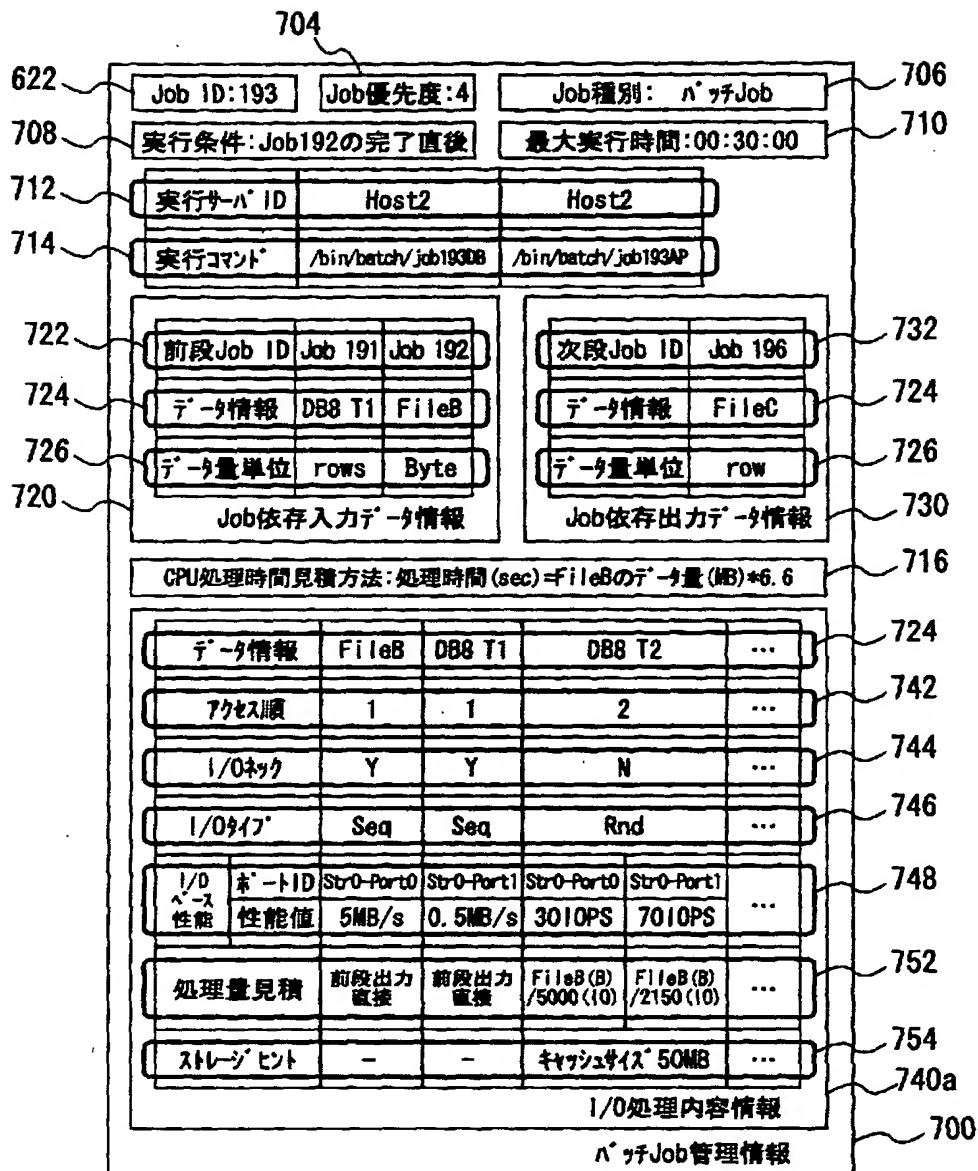
図 12

モニタ部位ID		Host0-Port0-LU0	Host0-Port0-LU0	...	514
モニタ内容		累積実行回数	累積データ転送量	...	516
前回取得情報	取得日時	'02/12/14 13:00:00	'02/12/14 13:00:00	...	518
	取得値	7907209	50298237893	...	
履歴内容		平均実行回数	平均データ転送量(MB/s)	...	522
履歴情報	サンプル0	期間	'02/12/1 00:00:00~ 00:05:00	'02/12/1 00:00:00~ 00:05:00	...
		平均	152	7.6	...
		最大	203	10.2	...
	サンプル1	期間	'02/12/1 00:05:00~ 00:10:00	'02/12/1 00:05:00~ 00:10:00	...
		平均	189	9.5	...
		最大	258	12.9	...
	⋮	⋮	⋮	⋮	⋮

510

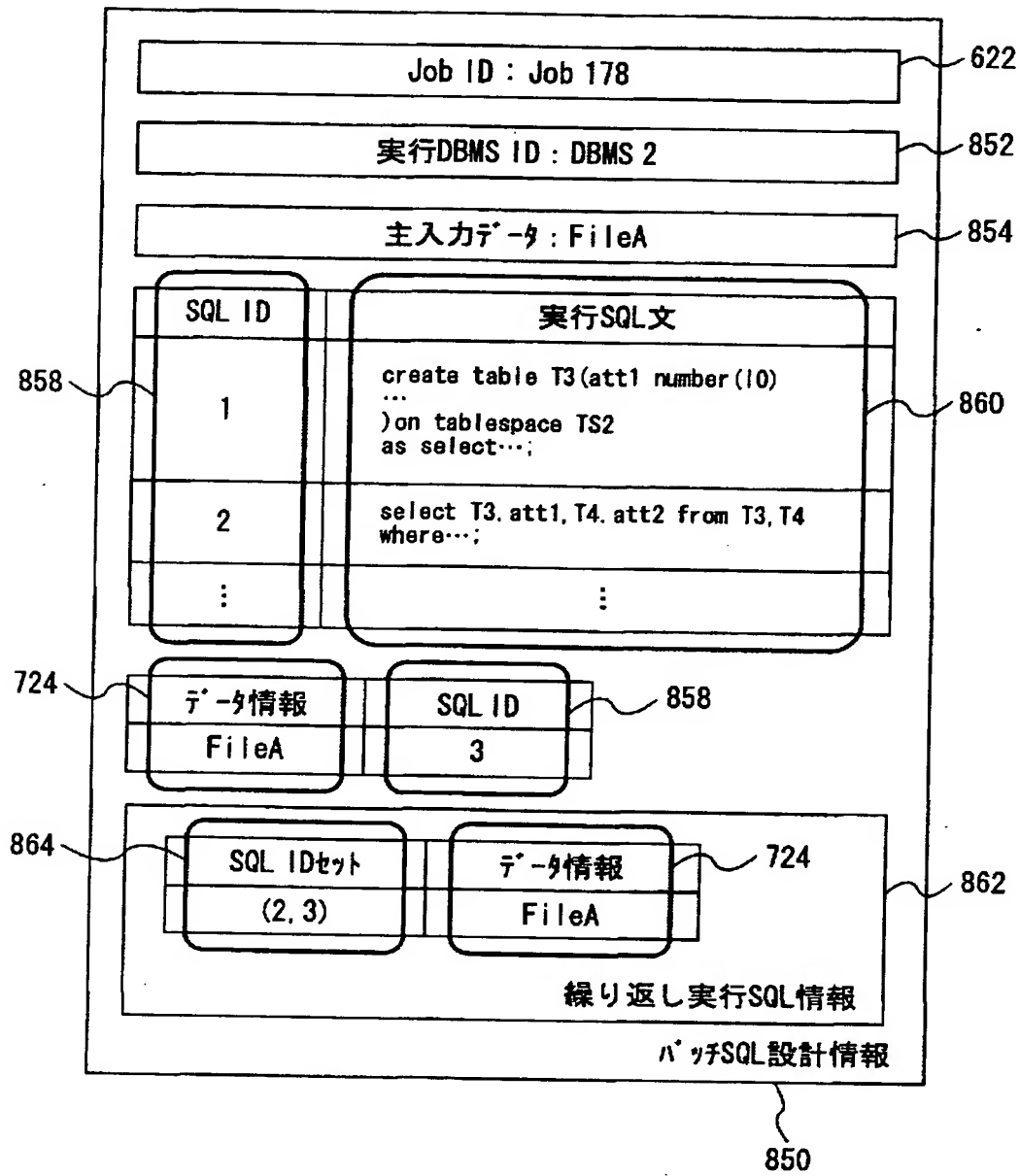
【図13】

図 13

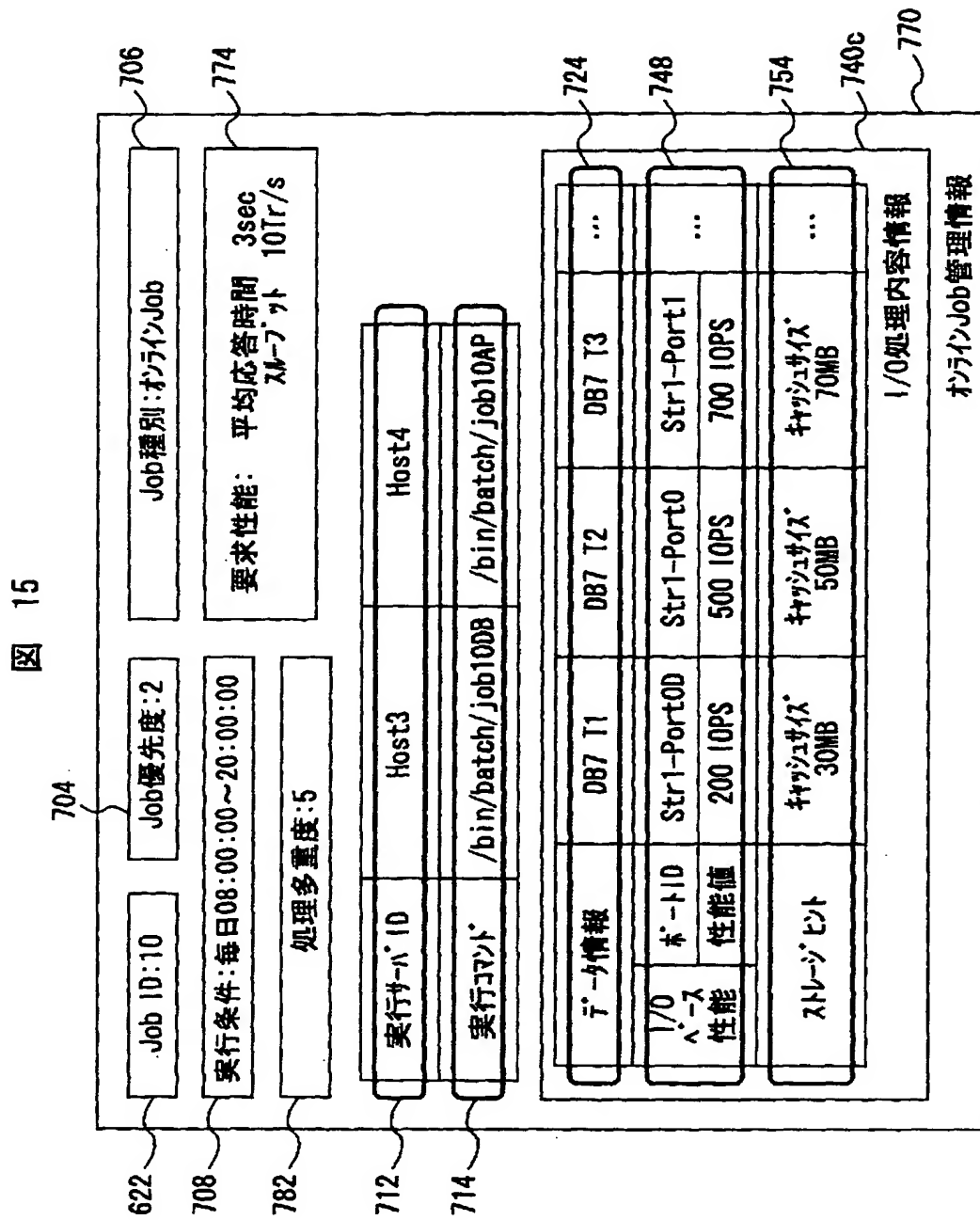


【図 14】

図 14



【図 15】



【図 1 6】

図 16

Jobポート設定情報					630
装置 ID	Storage 0	Storage 0	Storage 0	...	572
ポート ID	Str0-Port0	Str0-Port0	Str0-Port1	...	384
Job ID	J20	J41	J20	...	622
性能割合	20%	35%	30%	...	632
Jobポート設定情報					630
Jobキャッシュ量情報					650
装置 ID	Storage 0	Storage 0	Storage 0	...	572
キャッシュグループ ID	CGrp 1	CGrp 2	CGrp 2	...	462
Job ID	Job 31	Job 31	Job 41	...	622
割当情報	設定量	30MB	50MB	30MB	652
設定理由	初期設定	スループット 300 Tr/s	初期設定	スループット 300 Tr/s	652
Jobキャッシュ量情報					650
Jobリソース管理情報					620

【図 1 7】

図 17

880

Job優先度	1	2	3	4	5
余裕係数	1.2	1.15	1.1	1.05	1.05

704

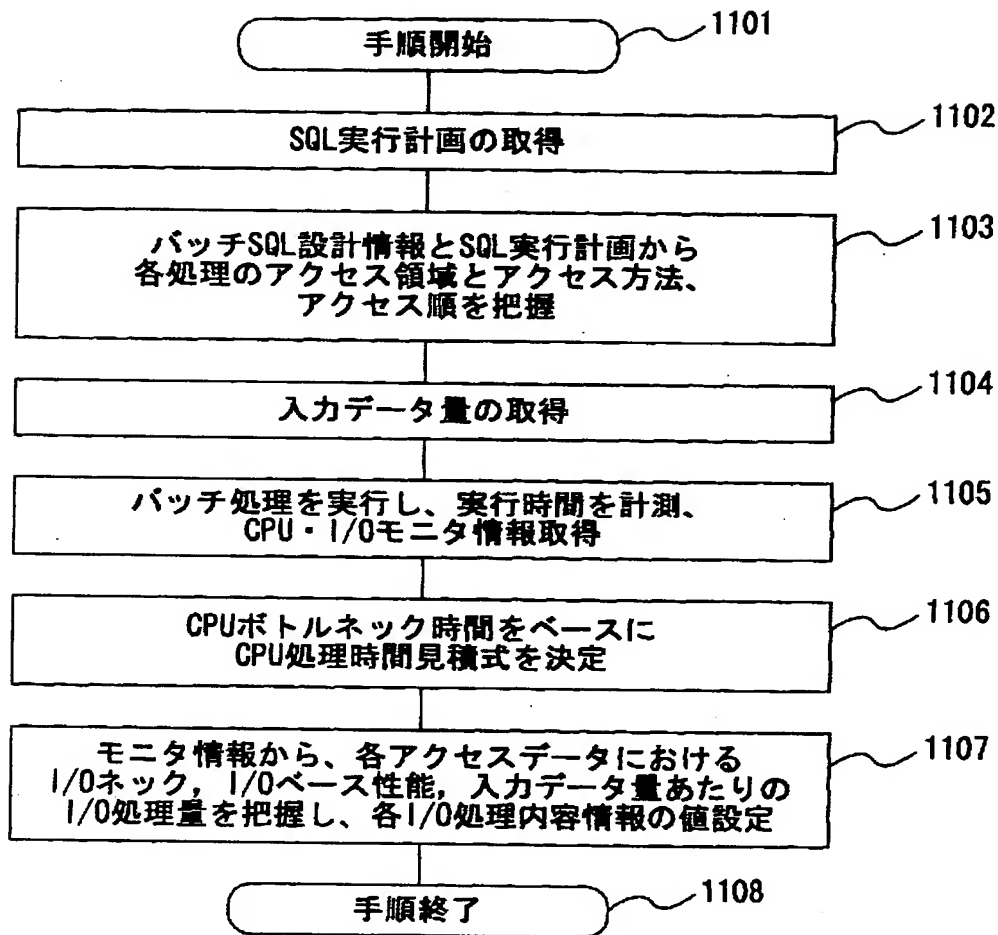
882

優先度制御情報



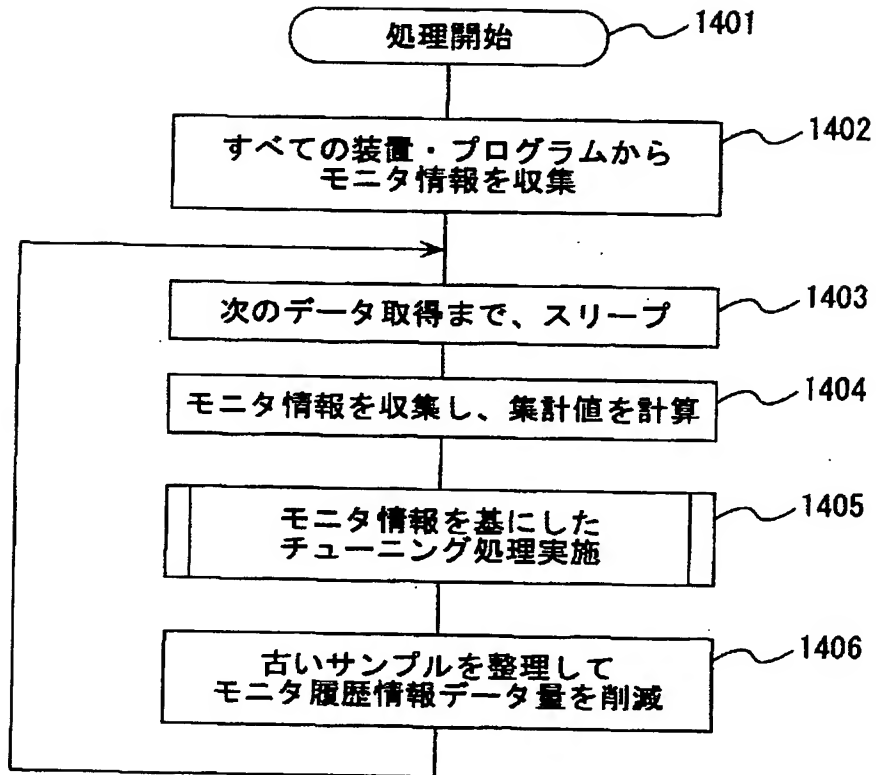
【図18】

図 18



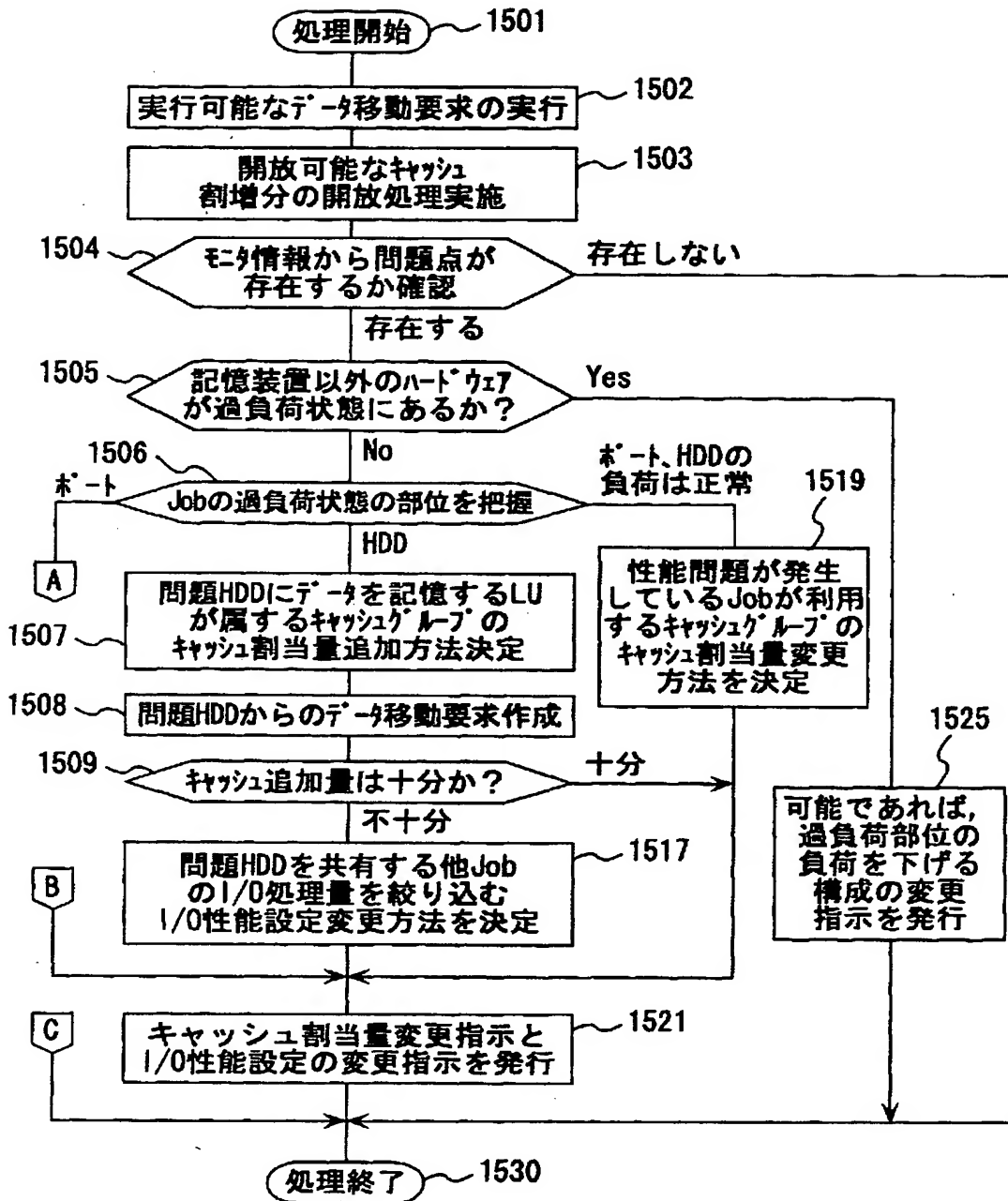
【図19】

図 19



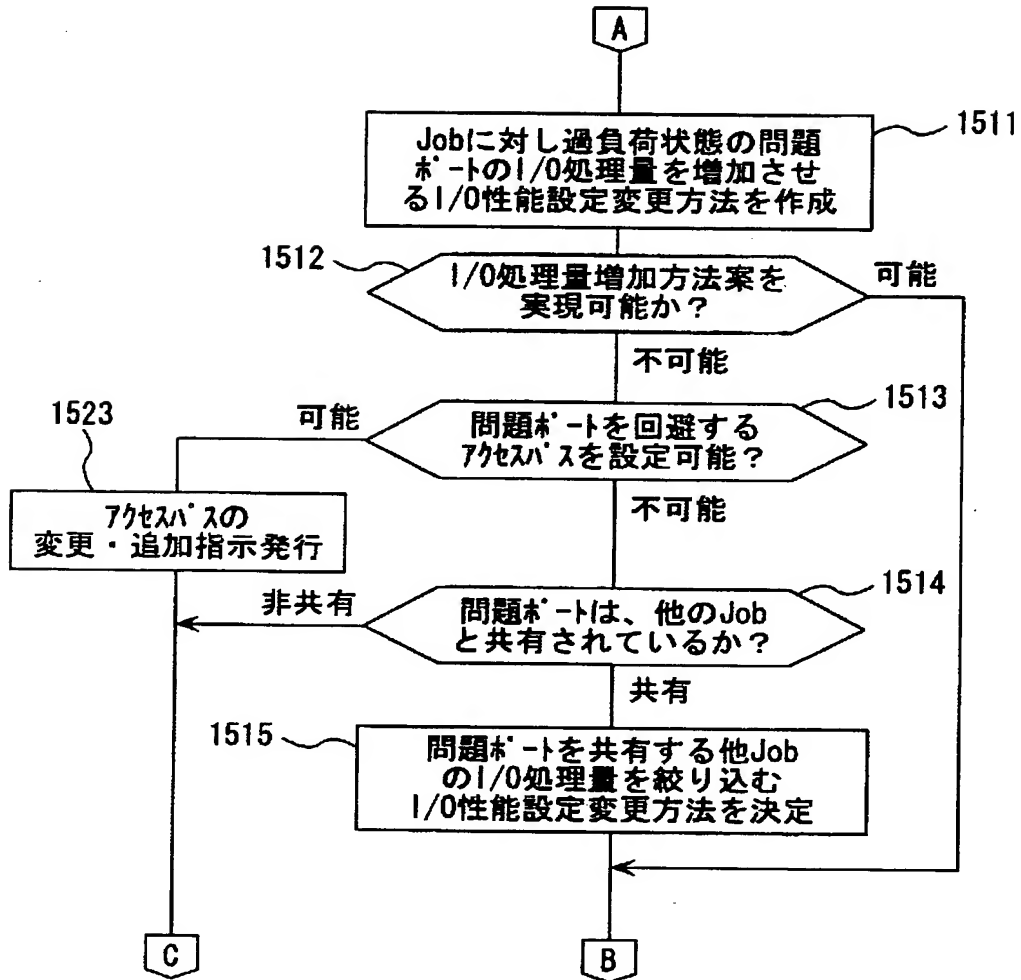
【図 20】

図 20



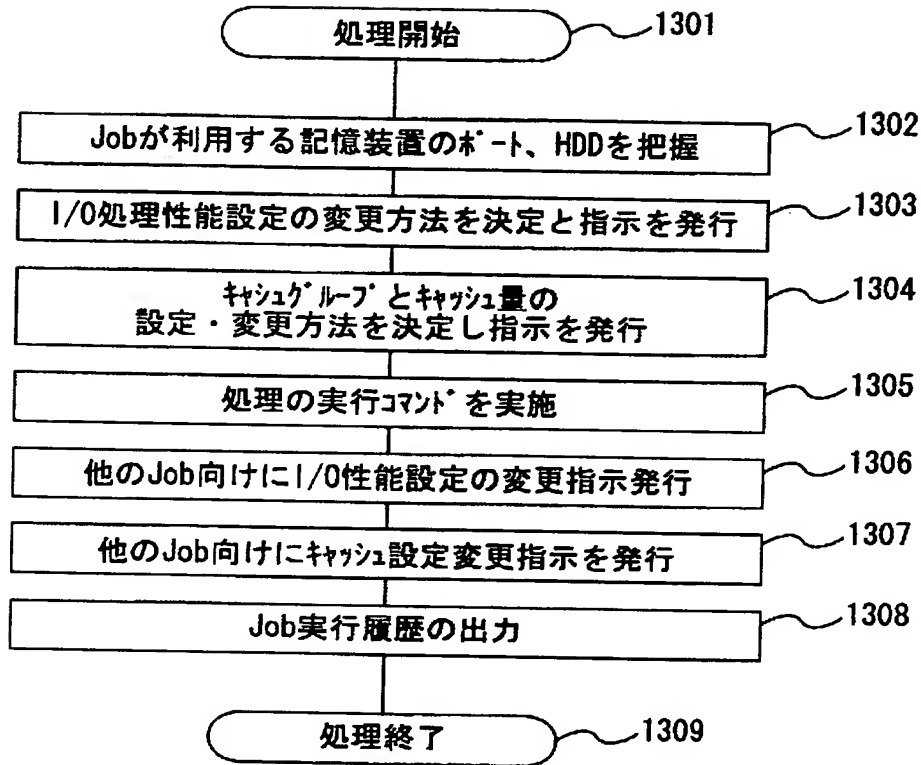
【図 21】

図 21



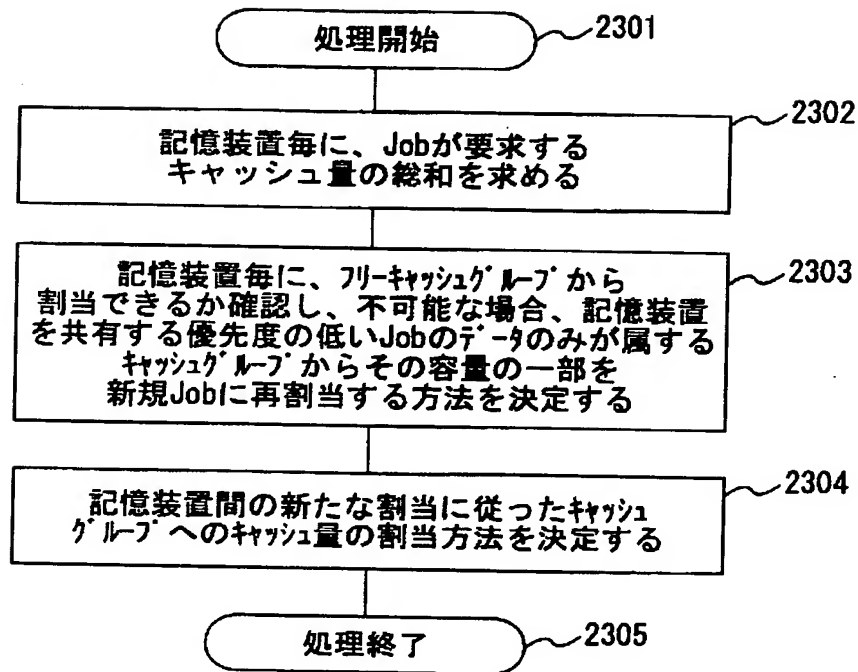
【図 2 2】

図 22



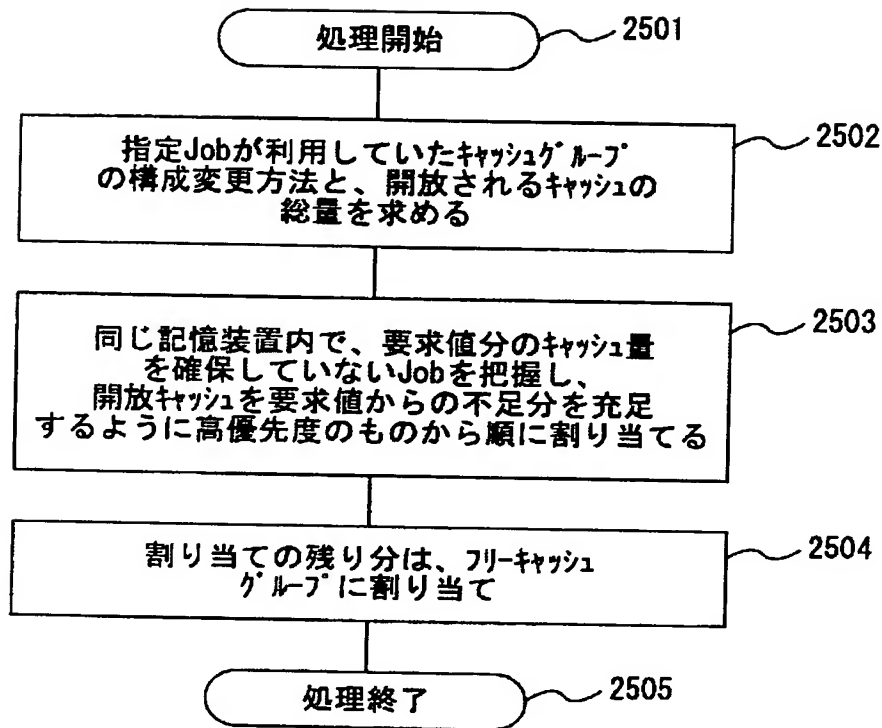
【図 2 3】

図 23



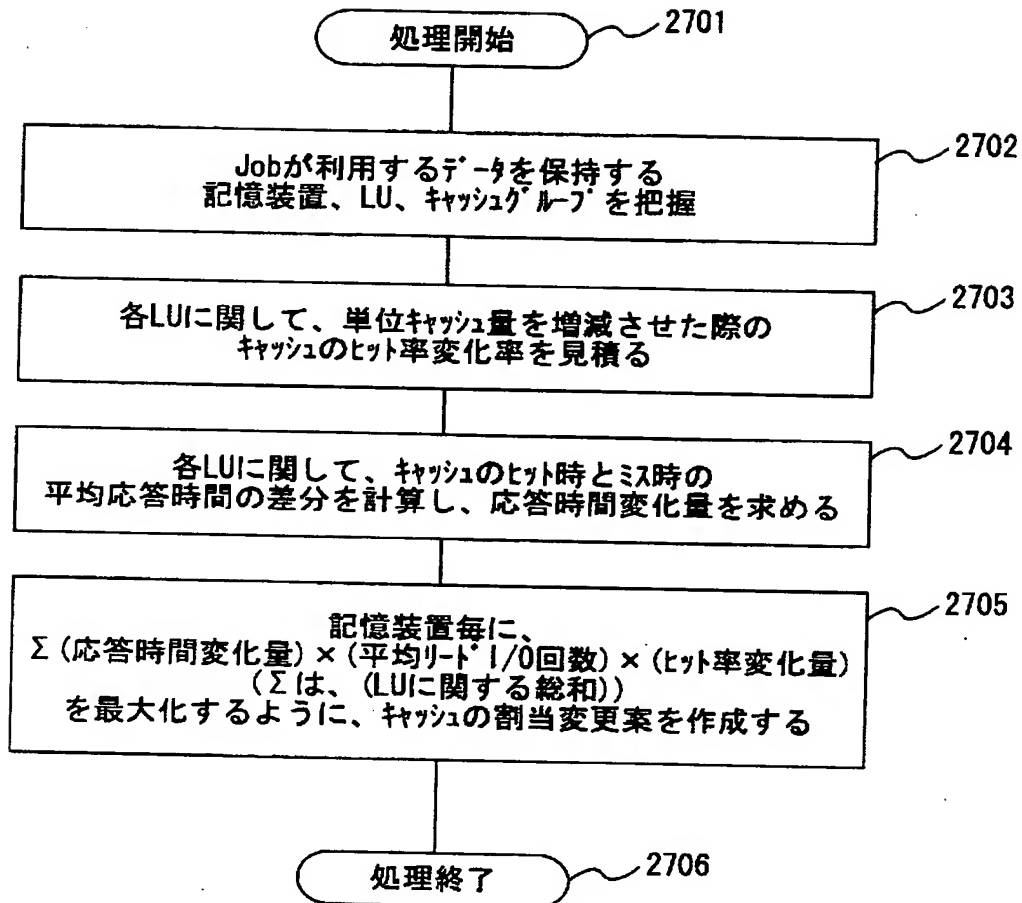
【図 2 4】

図 24



【図 2 5】

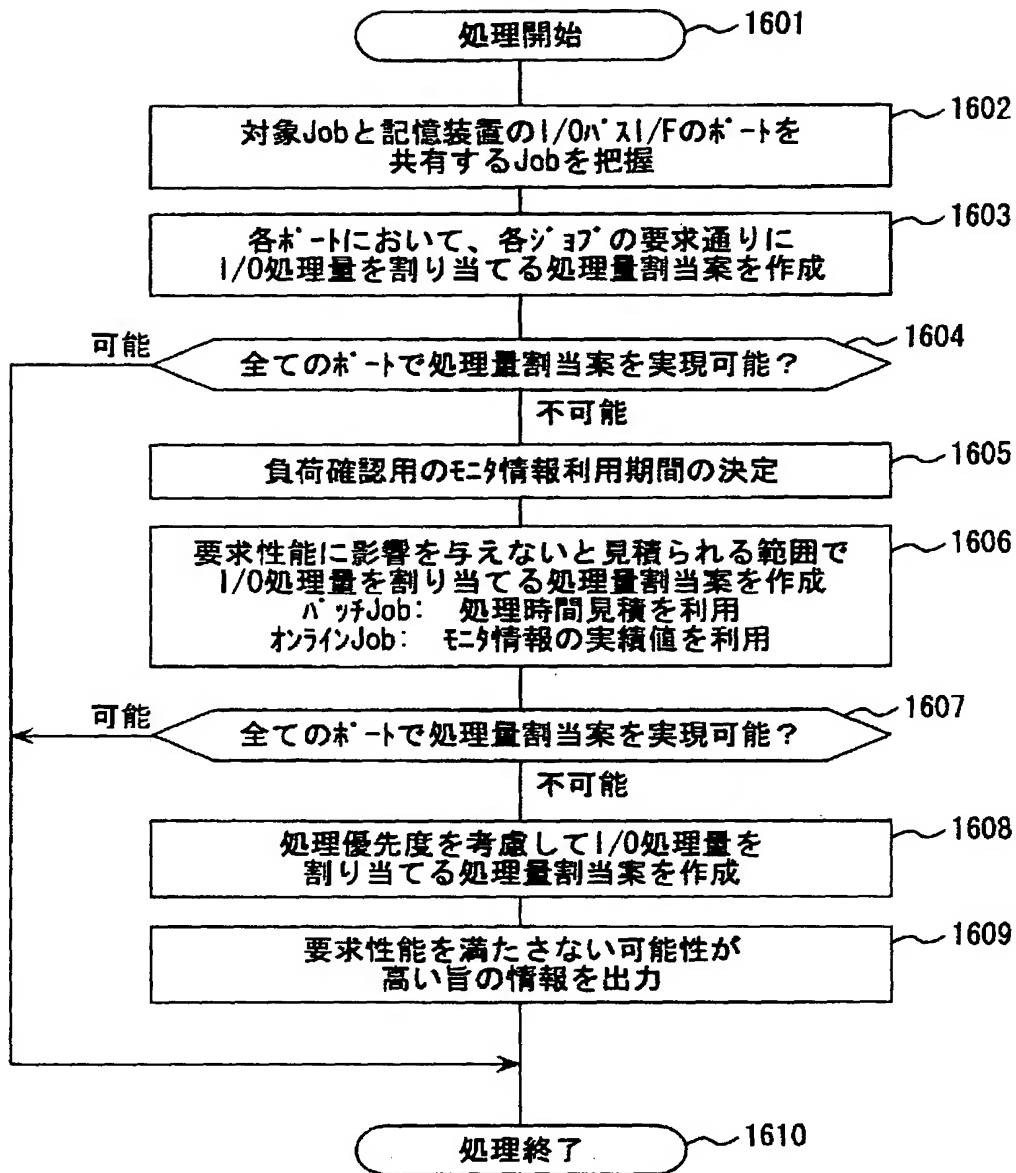
図 25





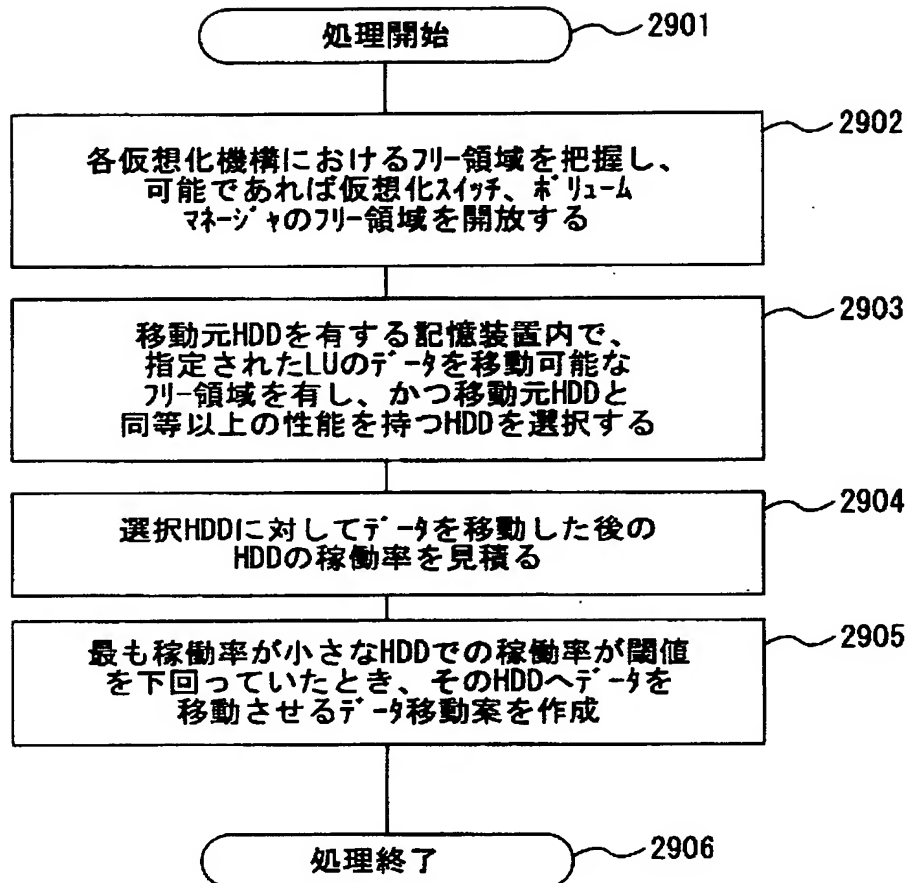
【図 26】

図 26



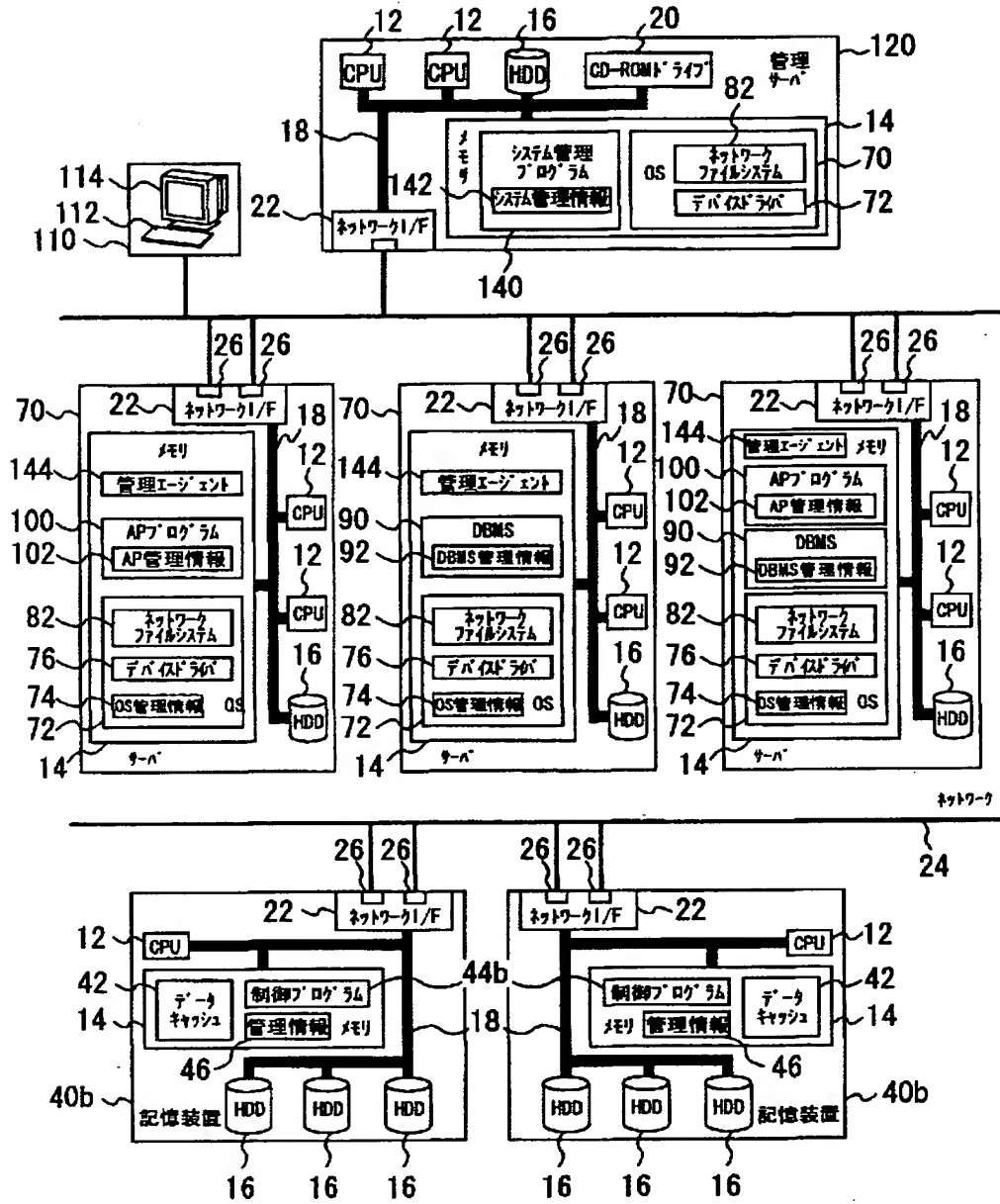
【図 2 7】

図 27



【図 28】

図 28



【書類名】 要約書

【要約】

【課題】 DBMS が稼動する計算機システムにおいて、記憶装置に対する性能管理を、ユーザの業務における性能指標を用いておこなえるようにし、性能管理を容易化する。

【解決手段】 計算機システムの管理サーバは、各構成要素の稼動状態や、業務の応答時間等をモニタし、要求性能等の予め与えた処理に関する情報や収集したモニタ情報を用いて、処理を実施するときやモニタ値からチューニングが必要と判断した場合に、ポートにおける処理量の割り当てや、データに対するキャッシュ量の割り当て、ディスク装置の論理構成等の変更指示を出す。

バッチ業務をおこなう処理に対しては、処理時間を見積る方法を管理サーバに与え、見積り実行時間を基に設定変更指示を出す。オンライン業務をおこなう処理では、処理の応答時間やスループットが要求性能に満たない場合に、負荷が高い部分の設定を変更する指示を出す。

【選択図】 図 1

出 願 人 履 歴 情 報

識別番号 [000005108]

1. 変更年月日 1990年 8月31日

[変更理由] 新規登録

住 所 東京都千代田区神田駿河台4丁目6番地  
氏 名 株式会社日立製作所